IoT Device Connection Efficiency Guidelines

Version 7.1

09 June 2021

This Industry Specification is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Antitrust Notice

This GSMA Permanent Reference Document (PRD) is classified by GSMA as an Industry Specification, as such it has been developed and is maintained by GSMA in accordance with the provisions set out GSMA AA.35 - Procedures for Industry Specifications.

Table of Contents

1 Introduction

1.1 Problem Statement

The predicted large scale growth of IoT Devices and their associated IoT Device Applications will create major challenges for Mobile Network Operators. One major challenge that Mobile Network Operators must overcome is the risk caused by the mass deployment of inefficient, insecure or defective IoT Devices on the Mobile Network Operators' networks. When deployed on a mass scale such devices can cause network signalling traffic to increase to a level which impacts network services for all users of the mobile network. In the worst cases the mass deployment of such IoT Devices can disable a mobile network completely.

Mobile Network Operators have faced similar issues in the past, most recently with the massive growth of smartphones. In this case many smartphone application developers inadvertently created many inefficient applications. Over the past decade Mobile Network Operators, smartphone device makers and smartphone application developers have worked together to resolve these difficulties through a mix of increasing network capacity (e.g. 3.5G and 4G network deployment), 3GPP standardisation, improvements to smartphone operating systems and development of smartphone application developer guidelines. With the forecasted high growth in IoT Devices the industry is in a similar situation to the start of the smartphone boom, but with a different group of device makers and application developers. With the IoT however the potential number of devices is higher and, due to the different commercial models for IoT Devices, it is far more challenging for the Mobile Network Operator to influence the behaviour of IoT Device manufacturers and IoT Device Application developers.

An IoT Device overusing the network may lead to problems such as:

- Reducing the lifetime of the (U)SIM card by increasing dramatically the read/write cycles.
- Increased power consumption of the device due to continuous restarts which may also affect the device lifetime.
- Local issues within the Mobile Network Operator's network such as cell congestion.
- Capacity and performance problems within the Mobile Network Operator's core network, such as signalling storms, which result in wide area network disruption.
- Negatively impacting the IoT Service's performance, potentially resulting in delayed communications, degradation of the service quality and even service outages.

IoT Devices overusing the mobile network can affect not only the devices causing the incident but also other devices on the same IoT Service Platform or those devices of other End Customers.

Network signalling resources are dimensioned assuming an overall device usage profile with a sensible balance between traffic and signalling needs. It is therefore important that IoT Devices using mobile networks adhere to some basic principles before they can be safely connected to mobile networks.

Good design is essential to ensure that IoT Device performance is optimized and to prevent failure mechanisms creating runaway situations which may result in network overload. In

situations where many IoT Devices of the same type may be deployed on a single mobile network the cumulative effect may have a detrimental impact on overall network performance. Poor design of IoT Device Application to IoT Service Platform communications which disregard the mobile network and IoT Device status may result in inefficient use of network and device resources, affecting the IoT Service experience end-to-end.

See Annex A for example cases where problematic IoT Device behaviour has impacted network and device performance.

1.2 Document Scope

In IoT scenarios IoT Device firmware and software play a significant part in determining the overall performance and behaviour of the IoT Service on the mobile network. With no human intervention to fall back upon, the mechanisms that manage recovery from IoT Service failure need to be built into IoT Devices.

This document will serve as a key deliverable from the GSMA Connected Living programme for 2014/15. The objective of this document is to specify requirements for efficient use of mobile network connectivity.

With the exception of section 9, the requirements and solutions captured in this document for efficient use of 3GPP mobile networks are for use within the current (short-term) timeframe, i.e. for the current generation of IoT Devices which do not necessarily support comparable 3GPP network efficiency features or are connecting to networks that do not support the necessary 3GPP network efficiency features.

In the mid to long term IoT Devices may make use of available features from 3GPP or other standards organisations to address the issues highlighted in this document. In section 9 we list the 3GPP feature that may be deployed within mobile networks and IoT Devices in the mid to long term.

1.3 Intended Audience

The target audiences for this document are Mobile Network Operators, IoT Service Providers, IoT Device makers, IoT Device Application developers, Communication Module Vendors and Radio Baseband Chipset Vendors.

1.3.1 Intended Use of the Document

1.3.1.1 Mobile Network Operators

The Mobile Network Operator shall promote the use of the requirements contained within this document. The Mobile Network Operator should make commercially reasonable efforts to reference this document in the connectivity contracts they agree with their IoT Service Providers.

1.3.1.2 IoT Service Providers

The IoT Service Provider shall ensure that their IoT Services and their IoT Device makers conform to the requirements stated within this document. The IoT Service Provider should reference this document in the supply contracts they place with their IoT Device makers.

1.3.1.3 IoT Device Maker

IoT Device makers are expected to implement the requirements contained within this document in the IoT Devices that they manufacture. The IoT Device maker will work with their IoT Application developer, Communication Module Vendor and Radio Baseband Chipset Vendor partners to implement the requirements contained within this document. The IoT Device maker should reference this document in the supply contracts they place with their IoT Application developer, Communication Module Vendor and Radio Baseband Chipset Vendor partners.

1.3.1.4 IoT Device Application Developer

The IoT Device Application developer shall ensure that their IoT Device Application conforms to the requirements stated within this document.

1.3.1.5 Communication Module Vendor

The Communication Module Vendor shall ensure that their Communication Modules conform to the requirements stated within this document.

1.3.1.6 Radio Baseband Chipset Vendor

The Radio Baseband Chipset Vendor shall ensure that their Radio Baseband Chipsets conform to the requirements stated within this document.

1.4 Key Words Used to Indicate Requirement Levels

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 (RFC2119)[2] (RFC8174)[17] when, and only when, they appear in all capitals, as shown here.

1.5 Definition of Terms

Term	Description
ADM	Access condition to an Elementary File (EF) which is under the control of the authority which creates this file
Back-off Timer	The Back-off Timer is a dynamic timer which value is based on a unique value for the device (desirably the IMSI) and the number of consecutive failures (which points to different Back-off Base Intervals).
End Customer	Means the consumer of IoT Services provided by the IoT Service Provider. It is feasible that the End Customer and IoT Service Provider could be the same actor, for example a utility company.
Fast Dormancy	Device power saving mechanism. See GSMA TS.18 [14].

Term	Description
Global Certification Forum	An independent worldwide certification scheme for mobile phones and wireless devices that are based on 3GPP standards. The GCF provides the framework within which cellular GSM, UMTS and LTE mobile devices and Communication Modules obtain certification for use on GCF Mobile Network Operators' networks. Obtaining GCF Certification on a mobile device ensures compliance with 3GPP network standards within the GCF Mobile Network Operators' networks. Consequently, GCF Mobile Network Operators MAY block devices from their network if they are not GCF certified. For more information, see http://www.globalcertificationforum.org
Internet of Things	The Internet of Things describes the coordination of multiple machines, devices and appliances connected to the Internet through multiple networks. These devices include everyday objects such as tablets and consumer electronics, and other machines such as vehicles, monitors and sensors equipped with machine-to-machine (M2M) communications that allow them to send and receive data.
IoT Communications Module	The IoT Communications component which provides wide area (2G, 3G, 4G & 5G) radio connectivity. Comprising of IoT Communications Module Firmware, Radio Baseband Chipset and UICC
IoT Communications Module Firmware	The functionality within the IoT Communications Module that provides an API to the IoT Device Application and controls the Radio Baseband Chipset.
IoT Device	A device, whose main function is to allow objects to be accessed, sensed and/or controlled remotely, primarily across existing mobile network infrastructures. An IoT Device consists of hardware and software that combine an IoT Device Application and a IoT Communications Module (see other definitions).
IoT Device Application	The application software component of the IoT Device that controls the IoT Communications Module and interacts with an IoT Service Platform via the IoT Communications Module.
IoT Device Host	The application specific environment containing the IoT Device e.g. vehicle, utility meter, security alarm etc.
IoT Server Application	An application software component that runs on a server and can exchange data and interact with the IoT Devices and the IoT Device Applications over the IoT Service Platform.
IoT Service	The IoT service provided by the IoT Service Provider.

Term	Description
IoT Service Platform	The service platform, hosted by the IoT Service Provider which communicates to an IoT Device to provide an IoT Service. The IoT Service Platform can exchange data with the IoT Device Application over the Mobile Network and through the Communication Module, using (among others) IP-based protocols over a packet-switched data channel. Also, the IoT Service Platform typically offers Device Management capabilities, acting as a so-called Device Management Server. Finally, the IoT Service Platform typically offers APIs for IoT Server Applications to exchange data and interact with the IoT Device Applications over the IoT Service Platform.
IoT Service Provider	The provider of IoT services working in partnership with a Mobile Network Operator to provide an IoT Service to an End Customer. The provider could also be a Mobile Network Operator.
Machine to Machine	Machine-to-Machine (M2M) is an integral part of the Internet of Things (IoT) and describes the use of applications that are enabled by the communication between two or more machines. M2M technology connects machines, devices and appliances together wirelessly via a variety of communications channels, including IP and SMS, to deliver services with limited direct human intervention turning these devices into intelligent assets that open up a range of possibilities for improving how businesses are run.
Mobile Network Operator	The mobile network operator(s) connecting the IoT Device Application to the IoT Service Platform.
PTCRB	The independent body established as the wireless device certification forum by North American Mobile Network Operators. The PTCRB provides the framework within which cellular GSM, UMTS and LTE mobile devices and Communication Modules obtain certification for use on PTCRB Mobile Network Operator networks. Obtaining PTCRB Certification on a mobile device ensures compliance with 3GPP network standards within the PTCRB Mobile Network Operators' networks. Consequently, PTCRB Mobile Network Operators MAY block devices from their network if they are not PTCRB certified. For more information, see http://ptcrb.com
Radio Baseband Chipset	The functionality within the IoT Communications Module that provides connectivity to the mobile network.
Requirements numbering:- TS34_X.X_REQ_YYY	TS.34 = this PRD number. X.X = the section number the requirement can be found in. REQ = Requirement YYY = the requirement number.
Subscriber Identity Module	Module provided by the Mobile Network Operator containing the International Mobile Subscriber Identity (IMSI) and the security parameters used to authenticate the (U)SIM with the Network. Seen as an authentication application contained in the Universal Integrated Circuit Card (UICC).

Term	Description
UICC	The smart card used by a mobile network to authenticate devices for connection to the mobile network and access to network services.

1.6 Abbreviations

Abbreviation	Description
3GPP	3rd Generation Project Partnership
API	Application Programming Interface
APN	Access Point Name
GCF	Global Certification Forum
GSM	Global System Mobile
GSMA	GSM Association
IMEI	International Mobile station Equipment Identity
IMSI	International Mobile Subscriber Identity
IoT	Internet of Things
IP	Internet Protocol
LTE	Long Term Evolution
M2M	Machine to Machine
NAT	Network Address Translation
OTA	Over The Air
PDP	Packet Data Protocol
PTCRB	A pseudo-acronym, originally meaning PCS Type Certification Review Board, but no longer applicable.
RFC	Request for Comments – a document of the Internet Engineering Task Force
RPM	Radio Policy Manager – see section 8
RRC	Radio Resource Control
SMS	Short Message Service
UMTS	Universal Mobile Telecommunications Service
(U)SIM	(Universal) Subscriber Identity Module
USB	Universal Serial Bus

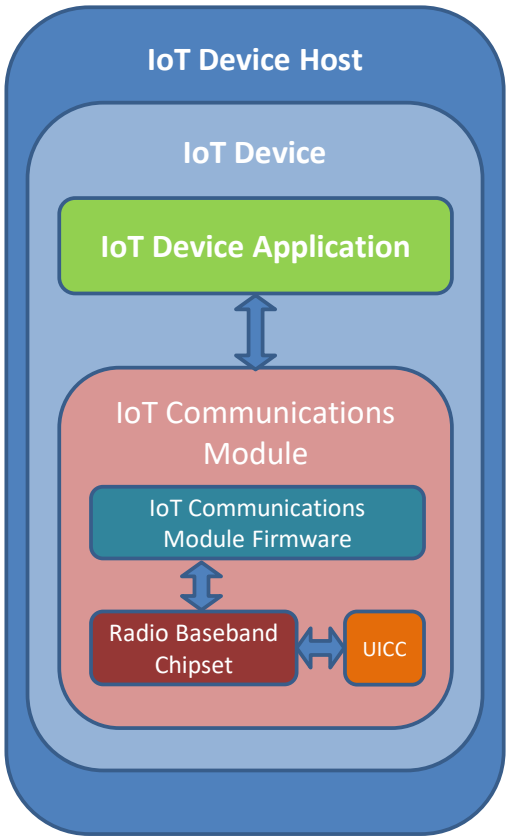
1.7 References

Ref	Document Number	Title
1	3GPP Specifications	www.3gpp.org
2	RFC 2119	Key words for use in RFCs to Indicate Requirement Levels http://www.ietf.org/rfc/rfc2119.txt
3	3GPP TS 36.331	Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification www.3gpp.org
4	3GPP TS 31.102	Characteristics of the Universal Subscriber Identity Module (USIM) application www.3gpp.org
5	GSMA SGP.02	Remote Provisioning Architecture for Embedded UICC Technical Specification www.gsma.com
6	3GPP TS 22.016	International Mobile station Equipment Identities (IMEI) www.3gpp.org
7	OMA DiagMon	OMA DiagMon Management Object Version 1.2 www.openmobilealliance.org
8	OMA DM	OMA Device Management Version 1.2 or 1.3 www.openmobilealliance.org
9	OMA FUMO	OMTA Firmware Update Management Object Version X.X www.openmobilealliance.org
10	GSMA TS.06	IMEI Allocation and Approval Process www.gsma.com
11	OMA ERELDDM_1.2	Enabler Release Definition for OMA Device Management www.openmobilealliance.org
12	3GPP TS 24.008	Mobile radio interface Layer 3 specification; Core network protocols; Stage 3 www.3gpp.org
13	3GPP TS 23.122	Non-Access-Stratus functions related to Mobile Station in idle mode www.3gpp.org
14	GSMA TS.18	Fast Dormancy Best Practices www.gsma.com
15	OMA LightweightM2M	OMA LightweightM2M www.openmobilealliance.org
16	GSMA IR.92	IMS Profile for Voice and SMS
17	RFC8174	Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words

2 IoT Architecture Assumptions (Informative Section)

2.1 Generalised IoT Device Architecture

In order to ensure a common vocabulary is used within this document an illustration of a generalised IoT Device architecture is shown in Figure 1 below.



IoT Device Host – The application specific environment containing the IoT Device e.g. vehicle, utility meter, security alarm etc.

IoT Device – The combination of both the IoT Device Application and the Communication Module.

IoT Device Application – The application software component of the IoT Device that controls the Communications Module and interacts with an IoT Service Platform via the communications module.

IoT Communication Module – The communications component which provides wide area (2G, 3G, 4G) radio connectivity. Comprising of IoT Communications Module Firmware, Radio Baseband Chipset and UICC

IoT Communications Module Firmware – The functionality within the IoT Communications Module that provides an API to the IoT Device Application and controls the Radio Baseband Chipset.

Radio Baseband Chipset – The functionality within the communications module that provides connectivity to the mobile network.

UICC – The smart card used by a mobile network to authenticate devices for connection to the mobile network and access to network services.

Figure 1: Generalised IoT Device Architecture

- IoT Device requirements can be found in section 3 of this document.
- IoT Device Application requirements can be found in section 4 of this document.
- Communication Module (and Radio Baseband Chipset) requirements can be found in sections 5, 7, 8 and 9 of this document.

2.2 Generalised IoT Service Architecture

Beyond the scope of the IoT Device itself, and considering the architecture of the end-to-end IoT Service, a generalised IoT Service Architecture can be described as follows:

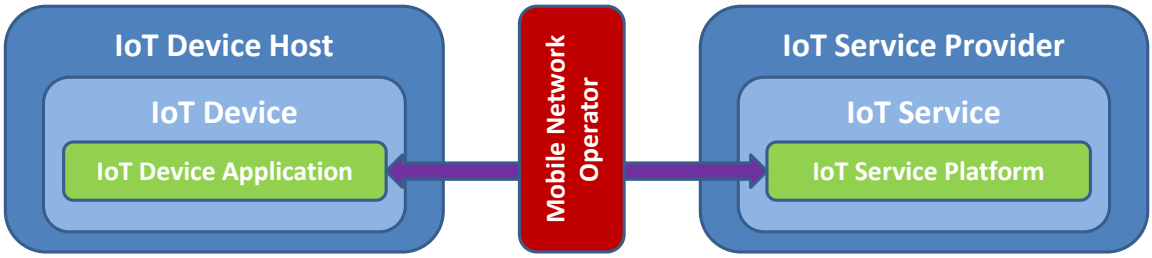


Figure 2: Generalised IoT Service Architecture

- **IoT Service Provider** – The provider of IoT services working in partnership with a network operator to provide an IoT Service to an End Customer. The provider could also be an MNO.
- **IoT Service** – The IoT service provided by the IoT Service Provider
- **IoT Service Platform** – The service platform, hosted by the IoT Service Provider which communicates to an IoT Device to provide an IoT Service.
- **Mobile Network Operator** – The mobile network operator(s) connecting the IoT Device Application to the IoT Service Platform

The IoT Service Platform very often exposes the deployed IoT devices and their data to applications located on the server side, e.g. in an enterprise system. These applications are the IoT Server Applications.

On the IoT Device, there is an evolution where the IoT Device Applications tend not to be monolithic, but are developed on top of a component providing several generic IoT functionalities (e.g. device management, security, location, application framework…) so as to focus on business-specific logic. This component is called the IoT Embedded Service Layer.

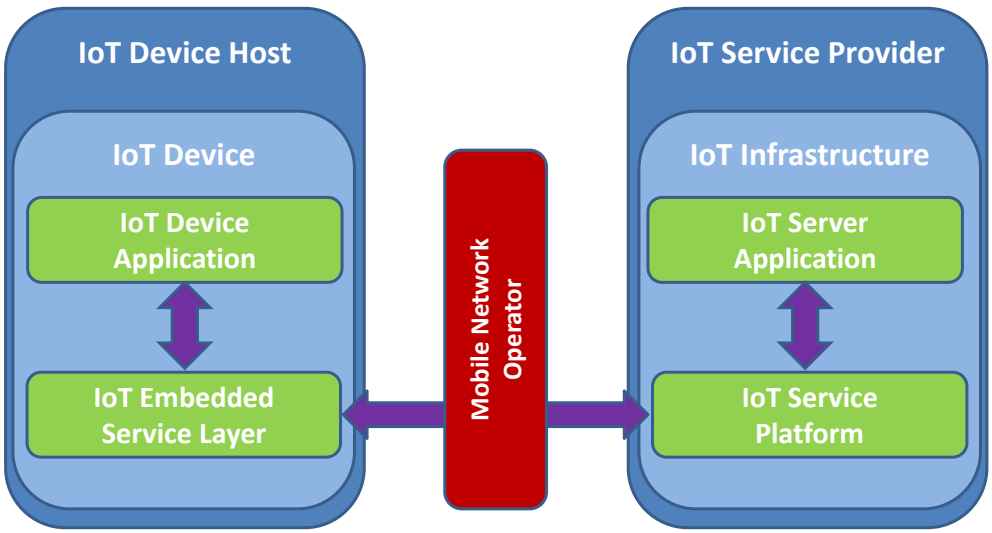


Figure 3: Generalised "Layered" IoT Service Architecture

- **IoT Server Application** – An application software component that runs on a server and exchanges data and can interact with the IoT Devices and the IoT Device Applications over the IoT Service Platform.
- **IoT Service Platform** – The service platform, hosted by the IoT Service Provider which communicates to an IoT Device to provide an IoT Service.
- **IoT Device Application** – The application software component of the IoT Device that controls the IoT Communications Module and interacts with an IoT Service Platform via the IoT Embedded Service Layer and the IoT Communications Module
- **IoT Embedded Service Layer** – The component offering generic IoT functionalities to IoT Device Application.

IoT Service Provider requirements can be found in section 6 of this document.

3 IoT Device Requirements (Normative Section)

TS.34_3.0_REQ_001	The IoT Device SHOULD conform to all IoT Device Application requirements defined in section 4
TS.34_3.0_REQ_002	The IoT Device SHALL conform to all Communication Module requirements defined in section 5.
TS.34_3.0_REQ_003	The IoT Device SHOULD conform to GSMA TS.24 “Operator Minimum Acceptance Values for Device Antenna Performance” [x].
TS.34_3.0_REQ_004	When required by the Mobile Network Operator, the IoT Device SHALL be certified by the GCF and/or the PTCRB.

4 IoT Device Application Requirements (Normative Section)

The requirements expressed in this section are targeted at the Device Application, as defined in the section 2.1 and 2.2 (figure 2) of the present document.

However, as shown on the figure 3 of the same section 2.2, the “software” running in an IoT Device is more and more split between:

- a generic M2M/IoT “IoT Embedded Service Layer”, offering generic IoT-related capabilities (such as security, connectivity management, subscription/notification mechanisms
- a business-specific “”IoT Device Application” that takes care only of the functionalities relevant for the customer business processes (eg. automotive system monitoring, industrial control, gas metering, etc.)

As a result, the IoT Device Application Requirements proposed in this section are considered fulfilled if:

- Either all of the requirements spelled out in the following section/
  4.0 – Monolithic IoT Device Application Requirements
  for “monolithic” IoT Device Applications such as pictured in the figure 2 of section 2.2
- Or all of the requirements spelled out in the following sections:

- 4.1 – Tiered IoT Device Application Requirements
  for “tiered” IoT Device Applications such as pictured in the figure 3 of section 2.2
- 4.2 – IoT Embedded Service Layer Requirements
  when such an embedded service layer is available in the IoT Device

The following table summarizes the 2 possible options to cover the IoT Device Application requirements:

Option 1 Application requirements are handled by the application only	**Option 2** Application requirements are handled by the Application and the Service Layer together	
Application requirements	**Application requirements**	**Service Layer Requirements**
TS.34_4.0_REQ_001	NA	TS.34_4.2_REQ_001
TS.34_4.0_REQ_002	NA	TS.34_4.2_REQ_002
TS.34_4.0_REQ_003	NA	TS.34_4.2_REQ_003
TS.34_4.0_REQ_004	NA	TS.34_4.2_REQ_004
TS.34_4.0_REQ_005	NA	TS.34_4.2_REQ_005
TS.34_4.0_REQ_006	NA	TS.34_4.2_REQ_006
TS.34_4.0_REQ_007	NA	TS.34_4.2_REQ_007
TS.34_4.0_REQ_008	NA	TS.34_4.2_REQ_008
TS.34_4.0_REQ_009	NA	TS.34_4.2_REQ_009
TS.34_4.0_REQ_010	TS.34_4.1_REQ_001	TS.34_4.2_REQ_010
TS.34_4.0_REQ_011	TS.34_4.2_REQ_002	TS.34_4.2_REQ_011
TS.34_4.0_REQ_012	NA	TS.34_4.2_REQ_012
TS.34_4.0_REQ_013	NA	TS.34_4.2_REQ_013
TS.34_4.0_REQ_014	NA	TS.34_4.2_REQ_014
TS.34_4.0_REQ_015	NA	TS.34_4.2_REQ_015
TS.34_4.0_REQ_016	NA	TS.34_4.2_REQ_016
TS.34_4.0_REQ_017	NA	TS.34_4.2_REQ_017
TS.34_4.0_REQ_018	TS.34_4.1_REQ_003	TS.34_4.2_REQ_018
TS.34_4.0_REQ_019	NA	TS.34_4.2_REQ_019
TS.34_4.0_REQ_020	TS.34_4.1_REQ_004	TS.34_4.2_REQ_020
TS.34_4.0_REQ_021	NA	TS.34_4.2_REQ_021
TS.34_4.0_REQ_022	NA	TS.34_4.2_REQ_022
TS.34_4.0_REQ_023	NA	TS.34_4.2_REQ_023
TS.34_4.0_REQ_024	NA	TS.34_4.2_REQ_024
TS.34_4.0_REQ_025	NA	TS.34_4.2_REQ_025
TS.34_4.0_REQ_026	NA	TS.34_4.2_REQ_026
TS.34_4.0_REQ_027	NA	TS.34_4.2_REQ_027
TS.34_4.0_REQ_028	NA	TS.34_4.2_REQ_028
TS.34_4.0_REQ_029	NA	TS.34_4.2_REQ_029

o Monolithic IoT Device Application Requirements (Normative Section)

TS.34_4.0_REQ_001	In the case of an IoT Device Application which needs to send data very frequently the IoT Device Application SHOULD use an “always-on” connectivity mechanism instead of activating and deactivating network connections (a ‘network connection’ being the establishment of a radio connection between the IoT Communications Module and the network) very frequently.
TS.34_4.0_REQ_002	The IoT Device Application SHOULD minimize the number of network connections between the IoT Device and the network. Data SHOULD be aggregated by the IoT Device Application into as big a chunk as possible before being compressed and sent over the communications network. If the IoT Device Application provides several IoT Services using the same IoT Communications Module, the IoT Device Application SHOULD coordinate each of the IoT Services network communication to make efficient use of the network.
TS.34_4.0_REQ_003	If permissible for the IoT Service, the IoT Device Application SHOULD avoid synchronized behaviour with other IoT Devices and employ a randomized pattern (e.g. over a period of time of a few minutes to several hours or days) for network connection requests.
TS.34_4.0_REQ_004	The IoT Device Application SHOULD be implemented securely. For example by following industry guidelines such as those provided by: • IETF – www.ietf.org • Open Web Application Security Project (OWASP) - www.owasp.org • W3C – www.w3.org • OASIS – www.oasis-open.org • OMA – www.openmobilealliance.org • 3GPP – www.3gpp.org • OneM2M – www.onem2m.org
TS.34_4.0_REQ_005	The IoT Device Application SHOULD implement appropriate security measures to prevent unauthorized or insecure device management functionality (e.g. diagnostics, firmware updates) of the IoT Device software and firmware. Such security measures SHALL apply to all local and remote (over the air) device management functionality.
TS.34_4.0_REQ_006	If the IoT Service requires the use of ‘keep alive’ messages, the IoT Device Application SHOULD automatically detect the Mobile Network Operator’s TCP_IDLE value or UDP_IDLE value (NAT timers) when using push services. This can be achieved by increasing the IoT Device Application’s polling interval until a network timeout occurs and then operating just below the timeout value. The IoT Device Application SHOULD adapt to the new value as opposed to using a hard coding a polling interval set within the device.

TS.34_4.0_REQ_007	If the IoT Service requires the use of 'keep alive' messages, use of dynamic polling interval (ref. TS.34_4.0_REQ_006) is preferred. However, if a fixed polling interval is used, the IoT Device Application SHOULD use a time value specified by the Mobile Network Operator. If the preferred value of the Mobile Network Operator is unknown a default value of 29 minutes is recommended as the polling interval when devices use TCP protocol. If a fixed polling interval is used, the IoT Device Application SHOULD allow remote and/or local configuration of the interval. Note: The suggested value of 29 minutes for devices using TCP protocol is recommended because the routers used by many Mobile Network Operators' will clear the Network Address Translation (NAT) entry for the IoT Device's data session 30 minutes after the last communication is sent to/from the IoT Device. Note: If the device uses UDP protocol the device must use a timer value appropriate for the target network operator environment.
TS.34_4.0_REQ_008	The IoT Device Application SHOULD be designed to cope with variances in mobile network data speed and latency considering the variety in performance of mobile communications technologies such as 2G, 3G and LTE.
TS.34_4.0_REQ_009	The IoT Device Application SHOULD be capable of adapting to changes in mobile network type and data speed at any given time.
TS.34_4.0_REQ_010	If data speed and latency is critical to the IoT Service the IoT Device Application SHOULD constantly monitor mobile network speed and connection quality in order to request the appropriate quality of content from the IoT Service Platform.
TS.34_4.0_REQ_011	The IoT Device Application SHOULD always be prepared to handle situations when communication requests fail. Communication retry mechanisms implemented within an IoT Device Application can vary and will depend on the importance and volume of downloaded data. Possible solutions can be: • Simple counting of failed attempts since the data connection was first established (often the easiest solution). • Monitoring the number of failed attempts within a certain period of time. For example, if the data connection is lost more than five times within an hour, then the request can be suspended. This can be a more reliable technique to avoid short but regular connection problems, such as when a device is moving away from one network cell to another. The data connection can be lost when the device switches between cells, but when the cell is providing good coverage; the request can be processed successfully. Depending upon the IoT Service, no communication request by the IoT Device Application SHOULD ever be retried indefinitely – the request SHOULD eventually timeout and be abandoned. Note: The requirements contained within section 5.2 of this document describe the functionality that, when implemented within the IoT Communications Module to monitor IoT Device Application behaviour, ensures the retry mechanisms implemented within the IoT Device Application do not prevent the normal operation of the mobile network.

TS.34_4.0_REQ_012	The IoT Device Application SHOULD monitor the number of network connections it attempts over a set period of time. If the number of connection attempts exceeds a maximum value the IoT Device Application SHOULD stop requesting network connectivity until the time period has expired. The maximum value SHALL be set by the IoT Service Provider. In the case the IoT Device exceeds the maximum value a report SHOULD be sent to the IoT Service Platform.
TS.34_4.0_REQ_013	The IoT Device Application SHOULD monitor the volume of data it sends and receives over a set period of time. If the volume of data exceeds a maximum value the IoT Device Application SHOULD stop sending and receiving data until the time period has expired. The maximum value SHALL be set by the IoT Service Provider. In the case the IoT Device exceeds the maximum value a report SHOULD be sent to the IoT Service Platform.
TS.34_4.0_REQ_014	The IoT Device Application SHOULD send a notification to the IoT Service Platform with relevant information when there is an unexpected power outage or unexpected battery power problem. This notification SHOULD follow the application scaling advice contained in Annex C.
TS.34_4.0_REQ_015	The IoT Device Application SHOULD use data transcoding and compression techniques, as per the intended QoS of the IoT Service, to reduce network connection attempts and data volumes.
TS.34_4.0_REQ_016	The IoT Device Application SHOULD be designed to ensure the application's network communication activity is not concentrated during periods of high network utilisation (i.e. utilises "off-peak" hours as guided by the Mobile Network Operator).
TS.34_4.0_REQ_017	The IoT Device Application SHOULD minimise any geographical network loading problems and tolerate any geographical network loading problems that MAY still occur.
TS.34_4.0_REQ_018	Each time there is a need to send data over the mobile network the IoT Device Application SHOULD classify the priority of each communication. For example, the IoT Device Application SHOULD distinguish between data that requires instantaneous transmission and delay tolerant data that could be aggregated and/or sent during non-peak hours.
TS.34_4.0_REQ_019	The IoT Device Application SHOULD not frequently reset the Communications Modem.
TS.34_4.0_REQ_020	When an IoT Device Application does not need to perform regular data transmissions and it can tolerate some latency for its IoT Service, it SHOULD implement a 'low power' mode where the device and its Communication Module is effectively powered down between data transmissions. This will reduce the power consumption of the IoT Device and reduce network signalling.
TS.34_4.0_REQ_021	Data sent from the IoT Device Application and the IoT Service Platform SHOULD be end-to-end encrypted to a security strength appropriate to the IoT Service. Note: It is recognised that for some IoT Services no encryption MAY be required.

TS.34_4.0_REQ_022	The IoT Device Application SHOULD authenticate the IoT Service Platform prior to data communication. The strength of authentication used SHOULD be appropriate to the IoT Service. Note: It is recognised that for some IoT Services no encryption MAY be required.
TS.34_4.0_REQ_023	VOID
TS.34_4.0_REQ_024	The IoT Device Application SHOULD support a “reset to factory settings” via remote and local connection.
TS.34_4.0_REQ_025	The IoT Device Application SHOULD support “time resynchronisation” via remote and local connection.
TS.34_4.0_REQ_026	If the IoT Device supports more than one family of communications access technology (for example 3GPP, TD-SCDMA, Wireless LAN) the IoT Device Application SHOULD implement a protection mechanism to prevent frequent ‘Ping-Pong’ between these different families of communications access technologies.
TS.34_4.0_REQ_027	For mass deployments of IoT Devices (e.g. >10,000 devices within the same mobile network), if the IoT Device supports more than one family of communications access technology (for example 3GPP, TD-SCDMA, Wireless LAN) the IoT Device Application SHOULD employ a randomised delay before switching to a different family of access technology.
TS.34_4.0_REQ_028	If the IoT Device contains a DHIR capable Communication Module (see Section 5.10) and the IoT Device leverages the Communication Module’s IMEI TAC the IoT Device Application SHALL report, via a secure method, the contents of the following custom nodes to the IoT Communications Module upon initial communication with the IoT Communications Module and at any time that any of the values of the custom node parameters change during the lifecycle of the IoT Device: • Host Device Manufacturer (see requirement TS.34_5.10_REQ_004) • Host Device Model (see requirement TS.34_5.10_REQ_005) • Host Device Software Version (see requirement TS.34_5.10_REQ_006) • Host Device Unique ID (see requirement TS.34_5.10_REQ_007) At minimum this includes IoT Device updates such as: • IoT Device firmware update by side-loading, USB, or other local methods; • IoT Device firmware update using a remote server.
TS.34_4.0_REQ_029	The IoT Device Application SHALL check that communication issues to the server are not caused by higher layer communications (like TCP/IP, UDP, ATM…) before starting to reset the communication module or re-establish the RRC Connection. Higher layers mechanisms SHALL then try to re-establish the connection with the server.

4.1 Tiered IoT Device Application Requirements (Normative Section)

TS.34_4.1_REQ_001	If data speed and latency is critical to the IoT Service the IoT Device Application SHOULD be able to retrieve mobile network speed and connection quality information from the IoT Embedded Service Layer in order to request the appropriate quality of content from the IoT Service Platform.
TS.34_4.1_REQ_002	The IoT Device Application SHOULD always be prepared to handle situations when communication requests fail, when such failure is reported by the IoT Embedded Service Layer.
TS.34_4.1_REQ_003	Each time there is a need to send data over the mobile network the IoT Device Application SHOULD classify the priority of each communication. For example, the IoT Device Application SHOULD distinguish between data that requires instantaneous transmission and delay tolerant data that could be aggregated and/or sent during non-peak hours. Such information about the priority of the communication SHOULD be communicated to the IoT Embedded Service Layer.
TS.34_4.1_REQ_004	When an IoT Device Application does not need to perform regular data transmissions and it can tolerate some latency for its IoT Service, it SHOULD communicate this information to the IoT Embedded Service Layer so that it can use this information in its interactions with the network..

4.2 IoT Embedded Service Layer Requirements (Normative Section)

TS.34_4.2_REQ_001	When supporting IoT Device Applications which need to send data very frequently the IoT Embedded Service Layer SHOULD use an “always-on” connectivity mechanism instead of activating and deactivating network connections (a ‘network connection’ being the establishment of a radio connection between the IoT Communications Module and the network) very frequently.
TS.34_4.2_REQ_002	The IoT Embedded Service Layer SHOULD minimize the number of network connections between the IoT Device and the network. Data SHOULD be aggregated by the IoT Embedded Service Layer into as big a chunk as possible before being compressed and sent over the communications network. The IoT Embedded Service Layer SHOULD coordinate each of the IoT Services network communication to make efficient use of the network.
TS.34_4.2_REQ_003	If permissible for the IoT Service, the IoT Embedded Service Layer SHOULD avoid synchronized behaviour with other IoT Devices and employ a randomized pattern (e.g. over a period of time of a few minutes to several hours or days) for network connection requests.

TS.34_4.2_REQ_004	The IoT Embedded Service Layer SHOULD provide security services to the IoT Device Application so as to provide a secure end-to-end service. For example by following industry guidelines such as those provided by: • IETF – www.ietf.org • Open Web Application Security Project (OWASP) - www.owasp.org • W3C – www.w3.org • OASIS – www.oasis-open.org • OMA – www.openmobilealliance.org • 3GPP – www.3gpp.org • OneM2M – www.onem2m.org
TS.34_4.2_REQ_005	The IoT Embedded Service Layer SHOULD enforce appropriate security measures to prevent unauthorized or insecure device management functionality (e.g. diagnostics, firmware updates) of the IoT Device software and firmware. Such security measures SHALL apply to all local and remote (over the air) device management functionality.
TS.34_4.2_REQ_006	If the IoT Service requires the use of 'keep alive' messages, the IoT Embedded Service Layer SHOULD automatically detect the Mobile Network Operator's TCP_IDLE value or UDP_IDLE value (NAT timers) when using push services. This can be achieved by increasing the IoT Device Application's polling interval until a network timeout occurs and then operating just below the timeout value. The IoT Embedded Service Layer SHOULD adapt to the new value as opposed to using a hard coding a polling interval set within the device.
TS.34_4.2_REQ_007	If the IoT Service requires the use of 'keep alive' messages, use of dynamic polling interval (ref. TS.34_4.2_REQ_006) is preferred. However, if a fixed polling interval is used, the IoT Embedded Service Layer SHOULD use a time value configurable by the Mobile Network Operator. If the preferred value of the Mobile Network Operator is unknown a default value of 29 minutes is recommended as the polling interval when devices use TCP protocol. If a fixed polling interval is used, the IoT Embedded Service Layer SHOULD allow remote and/or local configuration of the interval. Note: The suggested value of 29 minutes for devices using TCP protocol is recommended because the routers used by many Mobile Network Operators' will clear the Network Address Translation (NAT) entry for the IoT Device's data session 30 minutes after the last communication is sent to/from the IoT Device. Note: If the device uses UDP protocol the device must use a timer value appropriate for the target network operator environment.
TS.34_4.2_REQ_008	The IoT Embedded Service Layer SHOULD be designed to cope with variances in mobile network data speed and latency considering the variety in performance of mobile communications technologies such as 2G, 3G and LTE.
TS.34_4.2_REQ_009	The IoT Embedded Service Layer SHOULD be capable of adapting to changes in mobile network type and data speed at any given time.

TS.34_4.2_REQ_010	If data speed and latency is critical to the IoT Service the IoT Embedded Service Layer SHOULD constantly monitor mobile network speed and connection quality in order to request the appropriate quality of content from the IoT Service Platform.
TS.34_4.2_REQ_011	The IoT Embedded Service Layer SHOULD always be prepared to handle situations when communication requests fail. Communication retry mechanisms implemented within an IoT Device can vary and will depend on the importance and volume of downloaded data. Possible solutions can be: • Simple counting of failed attempts since the data connection was first established (often the easiest solution). • Monitoring the number of failed attempts within a certain period of time. For example, if the data connection is lost more than five times within an hour, then the request can be suspended. This can be a more reliable technique to avoid short but regular connection problems, such as when a device is moving away from one network cell to another. The data connection can be lost when the device switches between cells, but when the cell is providing good coverage; the request can be processed successfully. Depending upon the IoT Service, no communication request by the IoT Embedded Service Layer SHOULD ever be retried indefinitely – the request SHOULD eventually timeout and be abandoned. Note: The requirements contained within section 5.2 of this document describe the functionality that, when implemented within the IoT Communications Module to monitor IoT Embedded Service Layer behaviour, ensures the retry mechanisms implemented within the IoT Embedded Service Layer do not prevent the normal operation of the mobile network.
TS.34_4.2_REQ_012	The IoT Embedded Service Layer SHOULD monitor the number of network connections it attempts over a set period of time. If the number of connection attempts exceeds a maximum value the IoT Embedded Service Layer SHOULD stop requesting network connectivity until the time period has expired. The maximum value SHALL be set by the IoT Service Provider. In the case the IoT Device exceeds the maximum value a report SHOULD be sent to the IoT Service Platform.
TS.34_4.2_REQ_013	The IoT Embedded Service Layer SHOULD monitor the volume of data it sends and receives over a set period of time. If the volume of data exceeds a maximum value the IoT Embedded Service Layer SHOULD stop sending and receiving data until the time period has expired. The maximum value SHALL be set by the IoT Service Provider. In the case the IoT Device exceeds the maximum value a report SHOULD be sent to the IoT Service Platform.
TS.34_4.2_REQ_014	The IoT Embedded Service Layer SHOULD send a notification to the IoT Service Platform with relevant information when there is an unexpected power outage or unexpected battery power problem. This notification SHOULD follow the application scaling advice contained in Annex C.

TS.34_4.2_REQ_015	The IoT Embedded Service Layer SHOULD use data transcoding and compression techniques, as per the intended QoS of the IoT Service, to reduce network connection attempts and data volumes.
TS.34_4.2_REQ_016	The IoT Embedded Service Layer SHOULD be designed to ensure the application's network communication activity is not concentrated during periods of high network utilisation (i.e. utilises "off-peak" hours as guided by the Mobile Network Operator).
TS.34_4.2_REQ_017	The IoT Embedded Service Layer SHOULD minimise any geographical network loading problems and tolerate any geographical network loading problems that MAY still occur.
TS.34_4.2_REQ_018	Each time there is a need to send data over the mobile network the IoT Embedded Service Layer SHOULD take into account the information communicated by the IoT Device Application about the importance and urgency of the data (see requirement TS.34_4.1_REQ_003) so as to deliver the IoT Service without negatively impacting the network.
TS.34_4.2_REQ_019	The IoT Embedded Service Layer SHOULD not frequently reset the Communications Modem.
TS.34_4.2_REQ_020	When an IoT Device Application does not need to perform regular data transmissions and it can tolerate some latency for its IoT Service, the IoT Embedded Service Layer SHOULD implement a 'low power' mode where the device and its Communication Module is effectively powered down between data transmissions. This will reduce the power consumption of the IoT Device and reduce network signalling.
TS.34_4.2_REQ_021	Data sent from the IoT Embedded Service Layer and the IoT Service Platform SHOULD be end-to-end encrypted to a security strength appropriate to the IoT Service. Note: It is recognised that for some IoT Services no encryption MAY be required.
TS.34_4.2_REQ_022	The IoT Embedded Service Layer SHOULD authenticate the IoT Service Platform prior to data communication. The strength of authentication used SHOULD be appropriate to the IoT Service. Note: It is recognised that for some IoT Services no encryption MAY be required.
TS.34_4.2_REQ_023	VOID
TS.34_4.2_REQ_024	The IoT Embedded Service Layer SHOULD support a "reset to factory settings" via remote and local connection.
TS.34_4.2_REQ_025	The IoT Embedded Service Layer SHOULD support "time resynchronisation" via remote and local connection.
TS.34_4.2_REQ_026	If the IoT Device supports more than one family of communications access technology (for example 3GPP, TD-SCDMA, Wireless LAN) the IoT Embedded Service Layer SHOULD implement a protection mechanism to prevent frequent 'Ping-Pong' between these different families of communications access technologies.

TS.34_4.2_REQ_027	For mass deployments of IoT Devices (e.g. >10,000 devices within the same mobile network), if the IoT Device supports more than one family of communications access technology (for example 3GPP, TD-SCDMA, Wireless LAN) the IoT Embedded Service Layer SHOULD employ a randomised delay before switching to a different family of access technology.
TS.34_4.2_REQ_028	If the IoT Device contains a DHIR capable Communication Module (see Section **5.10**) and the IoT Device leverages the Communication Module's IMEI TAC the IoT Embedded Service Layer SHALL report, via a secure method, the contents of the following custom nodes to the IoT Communications Module upon initial communication with the IoT Communications Module and at any time that any of the values of the custom node parameters change during the lifecycle of the IoT Device: • Host Device Manufacturer (see requirement TS.34_5.10_REQ_004) • Host Device Model (see requirement TS.34_5.10_REQ_005) • Host Device Software Version (see requirement TS.34_5.10_REQ_006) • Host Device Unique ID (see requirement TS.34_5.10_REQ_007) At minimum this includes IoT Device updates such as: • IoT Device firmware update by side-loading, USB, or other local methods; • IoT Device firmware update using a remote server.
TS.34_4.2_REQ_029	The IoT Device Application SHALL check that communication issues to the server are not caused by higher layer communications (like TCP/IP, UDP, ATM…) before starting to reset the communication module which result in or re-establish the RRC Connection. Higher layers mechanisms SHALL then try to re-establish the connection with the server.

5 Communication Module Requirements (Normative Section)

5.1 Standards Compliance

TS.34_5.1_REQ_001	The IoT Communications Module SHALL be compliant with 3GPP specifications [1] unless otherwise stated within this document.
TS.34_5.1_REQ_002	The IoT Communications Module SHALL be certified by the GCF and/or the PTCRB.
TS.34_5.1_REQ_003	The IoT Communications Module SHALL investigate, and meet as required, the mobile network operator requirements for the target market(s).

5.2 Network Efficiency Requirements

TS.34_5.2_REQ_001	The IoT Communications Module requirements are as follows: 1. The IoT Communications Module SHALL support 3GPP Connection Efficiency features (as defined in section 9) within the Radio Baseband Chipset 2. The IoT Communication Module SHOULD support the Radio Policy Manager (as defined in section 8) within the Radio Baseband Chipset 3. The IoT Communication Module MAY support the Policy-Based Communication Efficiency Features (as defined in section 7) Note: Requirement TS.34_5.2_REQ_001-1 MAY require the target mobile network operator to have implemented the required 3GPP optional features in order to be effective.
TS.34_5.2_REQ_002	If the IoT Communications Module supports more than one family of communications access technology (for example 3GPP, TD-SCDMA, Wireless LAN) the device SHOULD implement a protection mechanism to prevent frequent 'Ping-Pong' between these different families of communications access technologies.
TS.34_5.2_REQ_003	The IoT Communication Module SHALL support the mechanism to control the number of RRC Connection Establishment and temporal offset for cell selection as defined in 3GPP TS36.331 [3]

5.3 IPv6 Requirements for Communication Modules that Support IPv6

The following requirements are only applicable to Communication Modules that support IPv6.

- The final target is IPv6 only connectivity, once most of the Internet will be IPv6.
- Remaining IPv4 services will be reachable through NAT64.
- Before IPv6 only connectivity stage is reached, a dual stack will be used to push migration towards IPv6.
- During the dual stack period, IPv4 rationalization solutions will be used.

TS.34_5.3_REQ_001	The IoT Communications Module SHOULD not send unsolicited messages (Router Solicitation for example).
TS.34_5.3_REQ_002	The IoT Communications Module SHOULD send only a AAAA DNS Query.
TS.34_5.3_REQ_003	The IoT Communications Module management system SHOULD be IPv6 based.
TS.34_5.3_REQ_004	The IoT Communications Module SHALL support the following IPv6 functionality: • Neighbour Discovery Protocol (apart from the exceptions noted in 3GPP TS 23.060 (3G) or TS 23.401 (LTE) • Stateless Address Auto Configuration • ICMPv6 protocol • IPv6 addressing architecture • IPv6 address text representation

TS.34_5.3_REQ_005	The IoT Communications Module SHOULD support the following IPv6 functionality: • Privacy Extensions for Stateless Address Auto-configuration in IPv6 • ROHC for IPv6 • IPv6 Router Advertisement Flags Options • Path MTU discovery • IPsec version 2 tunnel mode (IKE2)

5.4 Requirements for Communication Modules that Support LTE

The following requirements are only applicable to Communication Modules that support LTE.

TS.34_5.4_REQ_001	If voice calling over LTE is required by the IoT Service, the IoT Communication Module SHOULD support VoLTE (Voice over LTE) as per GSMA IR.92 [16].

5.5 Requirements for IoT Communication Modules that Support Fast Dormancy

The following requirements are only applicable to IoT Communication Modules that support Fast Dormancy.

TS.34_5.5_REQ_001	The Fast Dormancy algorithm within the IoT Communications Module SHOULD be triggered based on IoT Device data inactivity following suggested time parameters: • 5 to 10 (the specific value in range is to be defined by Mobile Network Operator) seconds for networks with PCH RRC State support (URA-PCH or Cell PCH) • Trigger disabled for networks without PCH RRC State support (URA-PCH or Cell PCH) The IoT Communications Module SHOULD ensure that background IP or IMS data flows would not be suspended by the Signalling Connection Release Indication (SCRI). Fast Dormancy best practices from GSMA TS.18 “Fast Dormancy Best Practices” [14] SHALL be followed.

5.6 (U)SIM Interface Requirements

TS.34_5.6_REQ_001	The IoT Communications Module SHALL support (U)SIM OTA management. See 3GPP TS31.102 [4]
TS.34_5.6_REQ_002	The IoT Communications Module SHOULD support remote provisioning as defined in GSMA SGP.01 “Remote Provisioning Architecture for Embedded UICC Technical Specification“ [5].

5.7 Security Requirements

TS.34_5.7_REQ_001	The IoT Communications Module SHALL implement a unique global IMEI and protect it against tampering. For details, please refer to 3GPP document TS 22.016 [6].

TS.34_5.7_REQ_002	The IoT Communications Module SHALL detect the removal of a powered UICC and terminate all network connections and services authenticated by the (U)SIM application on that UICC. Upon the removal of a powered UICC all temporary network authentication data related to the UICC SHOULD be deleted by the IoT Communications Module.
TS.34_5.7_REQ_003	The IoT Communications Module SHALL implement appropriate security measures to prevent unauthorized management (such as diagnostics, firmware updates etc.) of the IoT Communications Module.
TS.34_5.7_REQ_004	The IoT Communications Module SHALL implement a SIM lock function which allows the IoT Device to be locked to a specific UICC or range of UICCs. The state of the lock SHALL be remotely configurable.

5.8 Device Management

TS.34_5.8_REQ_001	The IoT Communications Module SHOULD support a standards based over the air device management protocol such as OMA DM [8] or OMA LightweightM2M [15].
TS.34_5.8_REQ_002	The IoT Communications Module SHOULD support a standards based firmware update mechanisms such as OMA FUMO [9].
TS.34_5.8_REQ_003	The IoT Communications Module SHOULD support a “reset to factory settings” via remote and local connection.
TS.34_5.8_REQ_004	The IoT Communications Module SHOULD support “time resynchronisation” via remote and local connection.

5.9 Subscription Identifier Requirements

Given the large potential number of IoT Devices, some national numbering and identification plans have been extended to avoid numbering exhaustion. The structure of these identifiers (MSISDN/Directory numbers, IMSIs) are defined in ITU-T Recommendations E.164 and E.212, and 3GPP TS 23.003.

TS.34_5.9_REQ_001	The IoT Communications Module SHALL support 15 digit Directory Numbers/MSISDNs.
TS.34_5.9_REQ_002	The IoT Communications Module SHALL support 2 and 3 digit based Mobile Network Codes IMSIs.

5.10 Requirements for Communication Modules that Support Device Host Identity Reporting (DHIR) (Normative Section)

As Communication Modules are certified for use on a network and integrated into various IoT Device Hosts the IMEI TAC range of the IoT Communications Module is often leveraged by the integrator of the IoT Device Host. For example, the PTCRB requirement is that not more than 10,000 units of the IoT Device Host can use the IMEI TAC range of the IoT Communications Module however it has frequently been seen that those rules are not always followed. In this situation the Mobile Network Operator has no traceability to the type of IoT Device Host that the IoT Communications Module is installed in and the number of those devices which are present on the network. This lack of traceability is problematic for several reasons including when field issues are discovered with a particular device and the

Mobile Network Operator is unable to pin point exactly what those devices are on its network.

This section defines the requirement for the IoT Communication Module to support a capability which reports IoT Device Host information.

This service utilizes a subset of the OMA Device Management standard. New custom OMA-DM nodes have been defined to collect the information from the IoT Device Host into which the IoT Communication Module is integrated.

It will be necessary for an MNO to define a server the OMA DM client will use to report this information to the network.

TS.34_5.10_REQ_001	The IoT Communications Module SHALL utilise the OMA DM specification [8] in order to implement the requirements within this section.
TS.34_5.10_REQ_002	The following standard nodes, as detailed in the OMA specification SHALL be supported by the IoT Communications Module in order for the MNO to gain visibility of the IoT Communications Module's detail and other pertinent Info. OMA Specification Support—OMA Device Management (DM) v1.2 or v1.3 The IoT Communications Module SHALL support OMA "Device Management" (DM) v1.2 or 1.3 specifications [8] and mandatory requirements contained within OMA "Enabler Release Definition for OMA Device Management" (ERELDDM_1.2) [11] for device provisioning/management.
TS.34_5.10_REQ_003	Support for IoT Device Host Reporting in the Device Detail Management Object For IoT Communications Modules embedded in an IoT Device Host, the IoT Device Host details SHALL be supported in an extension node within the Device Detail Management Object. These SHALL match the values for the associated PTCRB or GCF submission from requirement IDR4. The IoT Communications Module SHALL support four new custom nodes defined in TS.34_5.10_REQ_004,TS.34_5.10_REQ_005, TS.34_5.10_REQ_006 and TS.34_5.10_REQ_007.
TS.34_5.10_REQ_004	The following OMA-DM node has been defined to specify information related to the manufacturer of the IoT Host Device, this field will need to match the IoT Device Host manufacturer name that is referenced in the Mobile Network Operator lab certification of the IoT Device. **Type:** Host Device Manufacturer **Occurrence**: One **Format:** String **Name**: DevDetail/Ext/HostMan **Access Type**: GET The IoT Device Host manufacturer will be maintained in the node by the IoT Communications Module OMA DM client.
TS.34_5.10_REQ_005	The following OMA-DM node has been defined to specify the Model name/number of the IoT Device Host. This SHALL match the model name/number used in the certification of the IoT Device.

	Type: Host Device Model **Occurrence:** One **Format**: String **Name:** DevDetail/Ext/HostMod **Access Type**: GET The IoT Host Device model will be maintained in the node by the IoT Communication Module OMA DM client.
TS.34_5.10_REQ_006	The following OMA-DM node has been defined to specify the software version of the IoT Device Host, this information SHALL be populated by the IoT Device Host manufacturer, SHALL match the version of SW certified by PTCRB and must be updated whenever the SW is updated on the device. **Type**: Host Device Software Version **Occurrence:** One **Format**: String **Name**: DevDetail/Ext/HostSwV **Access Type**: GET The IoT Host Device software version will be maintained in the node by the IoT Communication Module OMA DM client
TS.34_5.10_REQ_007	The following OMA-DM node has been defined to specify the unique ID allocated to the IoT Device Host by the certifying Mobile Network Operator. Mobile Network Operators' MAY decide to include this field if they need a way to monitor for uncertified devices used on the network. **Type:** Host Device Unique ID **Occurrence:** One **Format:** Alphanumeric String **Name:** DevDetail/Ext/HostUniqueID **Access Type:** GET The IoT Device Host Unique ID is assigned by the Mobile Network Operator and will be stored in this node.
TS.34_5.10_REQ_008	**Interface Between IoT Communications Module and IoT Device Host** The Communication Module manufacturer SHALL provide a mechanism for the IoT Device Host to populate the information into the custom nodes (TS.34_5.10_REQ_001 ~ TS.34_5.10_REQ_007). It is at the Communication Module manufacturer's discretion to determine how to make the fields available to the host manufacturer to populate. This interface must be a secure interface which cannot be subject to reverse engineering or monitoring such that the content identifying the host device to cannot be compromised and potentially utilized to create cloned host devices utilizing a similar IMEI TAC range.
TS.34_5.10_REQ_009	**Device Description Framework Submission** The IoT Communications Module manufacturers SHALL submit the Device Description Framework (DDF) for the IoT Communications Module to the Mobile Network Operator. IoT Communications Module manufacturers SHALL ensure that the DevDetail, DevInfo and DM Account objects reflect the actual properties and information in use in the IoT Communications Module.
	Device Management Bootstrap DM Server Settings

TS.34_5.10_REQ_010	The IoT Communications Module SHALL support the factory loading of DM Server settings that are required to connect to the MNO DM server. The IoT Communications Module manufacturer SHALL obtain the most current values from the MNO and configure these into the module before shipping them to distribution channels. If multiple MNOs are to be supported by a common module the IoT Communications Module supplier SHOULD implement a methodology to differentiate MNO DM server settings based on the MNO of the UICC.
TS.34_5.10_REQ_011	**[DMBOOT] Complete Setup Option using NETWPIN** The Bootstrap process SHALL use NETWPIN, and devices SHALL not prompt the user with a confirmation prompt to complete the set up.
TS.34_5.10_REQ_012	**[DMBOOT]- DM Accounts** IoT Communications Modules SHALL support only 3 DM Accounts per MNO.
TS.34_5.10_REQ_013	**[DMBOOT]- Expose Factory Bootstrap Account Parameters on the Device** To facilitate troubleshooting during the testing process the IoT Communications Module manufacturer SHALL provide a means of exposing the factory bootstrap account parameters on the module. This SHALL be provided via a means to which the tester can select and read (but not modify) the parameters in each factory bootstrap account. Another means would be for the module manufacturer to provide a device utility.
TS.34_5.10_REQ_014	**DM Client support for Nonce Resynchronization** The DM client that uses MD5 or HMAC authentication for security must support client initiated nonce resynchronization. This is required SHOULD the nonce value become stale. The module manufacturer SHALL use the same authentication type on the module during IOT and production server testing and throughout the life of the device.
TS.34_5.10_REQ_015	**Device Management Protocol v1.2 or v.1.3** The IoT Communications Module SHALL support all mandatory requirements of [DMPRO_1.2] or [DMPRO_1.3].
TS.34_5.10_REQ_016	**Generic Alert—DM 1.2 or 1.3** The IoT Communications Module SHALL support the generic alert capabilities specified in [DMPRO_1.2] or [DMPRO_1.3].
TS.34_5.10_REQ_017	**Device Management Tree and Descriptions DM 1.2/1.3 - TStamp Support** In addition to the mandatory properties of nodes, the IoT Communications Module SHALL also support the TStamp property.
TS.34_5.10_REQ_018	**Device Management Tree and Descriptions DM 1.2/1.3 - VerNo Support** In addition to the mandatory properties of nodes, the IoT Communications Module SHALL also support the VerNo property.
TS.34_5.10_REQ_019	**Management Tree Requests - TNDS Attribute** The IoT Communications Module SHALL support requests for a part of a management tree using the Struct attribute. Requests of the form: Get <URI>?list=TNDS

	where <URI> is any subset of the management tree including the root SHALL be supported.
TS.34_5.10_REQ_020	**MIME Type - WBXML Encoded Management Objects** The IoT Communications Module SHALL support the MIME type application/vnd.syncml.dmddf+wbxml and associated WBXML encoded management objects [DMTND_1.2] or [DMTND_1.3].
TS.34_5.10_REQ_021	The following OMA-DM node has been defined to specify the IMEI SV for the IoT Communications Module. Mobile Network Operators' MAY decide to include this field if they need a way to monitor for uncertified devices used on the network. **Type:** IMEI SV Occurrence **Occurrence:** One **Format:** Numeric String (2 digit SV) **Name:** DevDetail/Ext/IMEISV **Access Type:** GET The IoT Communications Module IMEI is reported in DevInfo/DevId with the SV to be stored in the IMEI SV node.
TS.34_5.10_REQ_022	**Support for Operating System Details in the Device Detail Management Object** The current Operating System details for the IoT Communications Module SHALL be reported in an extension node within the Device Detail Management Object. **Type:** Operating System Name. For example: Android. **Occurrence:** One **Format:** String **Name:** DevDetail/Ext/OSName **Access Type:** GET **Type:** Operating System Version. For example: 4.4 **Occurrence:** One **Format:** Numeric String **Name:** DevDetail/Ext/OSVersion **Access Type:** GET
TS.34_5.10_REQ_023	**or 1.3** The IoT Communications Module SHALL support the DevInfo, DevDetail and DMAcc objects as mandated in [DMSTDOBJ_1.2] or [DMSTDOBJ_1.3].
TS.34_5.10_REQ_024	**Device Management Notification—DM 1.2 or 1.3** The IoT Communications Module SHALL support notification as specified in [DMNOTI_1.2] or [DMNOTI1.3]. Note that features of sections 5 and 6 of [DMNOTI_1.2] or [DMNOTI_1.3] are mandatory.
TS.34_5.10_REQ_025	**GET Default APN** The IoT Communications Module SHALL include the module default APN in the response to the OMA DM GET (device details). Note: the ModifiedTimeStamp field and value SHALL be included in the Extra node of each setting to indicate when the setting was modified (using UTC). If this field is absent, then the setting was not changed, and remains the factory setting.

TS.34_5.10_REQ_026	**REPLACE Default APN** The IoT Communications Module SHALL immediately replace the default APN after it has completed the OMA DM REPLACE command to replace APN, and SHOULD not require a module power cycle or reset. The default APN MAY have multiple instances stored in different memory areas of the module, all instances SHALL be replaced. APN replacement SHOULD not require user validation or acknowledgement. The new APN SHALL persist through power cycle. The new APN SHALL persist through factory reset of the device.
TS.34_5.10_REQ_027	**ADD Default APN** Typically the Device Management server assumes the module already has management nodes for managing the default APN, so it would attempt to send a REPLACE command to replace the default APN. If that SHOULD fail, then it tries to send the ADD command with the new value of the default APN. The ADD command for the following targets SHOULD be interpreted as adding new management nodes on the module to manage the default APN. Subsequent REPLACE command to these nodes SHALL affect the default APN. The added management nodes do not need to persist through factory reset, but they must persist through power cycle. The APN change resulting from the ADD command SHALL persist through power cycle and factory reset. Adding default APN management nodes SHOULD not require user validation or acknowledgement. The add command SHALL take effect immediately after the command is complete, and SHOULD not require a device power cycle or device reset.
TS.34_5.10_REQ_028	**IoT Communications Module Initiated Update—Generic Alert** For IoT Communications Module initiated updates, modules SHALL use the Generic Alert format for the update request sent to the server.
TS.34_5.10_REQ_029	**IoT Communications Module Initiated Session following a non-FOTA update** IoT Communications Modules which are updated using one of the following scenarios SHALL automatically initiate a session with the Device Management platform to report device details from the Device Detail Management Object new device details following the update. This is needed to keep back-end systems in sync with the new device details. • Module update by sideload/USB • Module update using a proprietary OEM Device Management server The details from the Device Detail Management Object reported to the Device Management server SHALL include at minimum the following: • IMEI • Current Firmware version • Actual WLAN MAC address (not the default WLAN MAC address) • Original Firmware version • Previous Firmware version • Date stamp for initial activation of the device • Date stamp for last software update on the device
TS.34_5.10_REQ_030	**IoT Communications Module Initiated Update—Alert Type**

	For IoT Communications Module initiated updates, devices SHALL use the OMA FUMO alert type "org.openmobilealliance.dm.firmwareupdate.devicerequest".
TS.34_5.10_REQ_031	**IoT Communications Module Initiated Update—URI** For IoT Communications Module initiated updates, the URI in the alert message sent by the module must point to the dynamic node representing a single firmware update management object in the tree.
TS.34_5.10_REQ_032	**IoT Communications Module Initiated Update—Data** For module initiated updates, the data element SHALL be included in the alert message to indicate the implementation details.

TS.34_5.10_REQ_033	**Support Secure Technology for End-2-End Connections in DHIR**
	
	Summary: The secure connection technology must meet contemporary and evolving requirements for authentication and data privacy over the targeted end-to-end connection within the scope of this requirement.
	
	• Authentication of the server by the client device must be supported by way of X.509 public key technologies, commonly known as "certificates".
	• Authentication of the client by the server is permitted.
	• Secure transport protocol must include TLS 1.0 and TLS 1.1.
	  o Secure transport protocol support for TLS 1.2 is strongly recommended
	• Secure transport protocol SHOULD not support any version of SSL.
	• The cipher suite used for data encryption SHOULD be based on contemporary, strong ciphers as commonly supported in TLS 1.0 or greater
	  o Support for TLS 1.2 is strongly recommended.
	• Certificates MAY be issued by a certificate authority of the carrier's choice.
	• Certificates SHOULD abide by contemporary standards for signature strength.
	• No IP address SHALL be used in the bootstrap account for the server URL
	Only FQDN SHALL be used in the bootstrap account for the server URL for an https connection

6 IoT Service Provider Requirements (Normative Section)

TS.34_6.0_REQ_001	If permissible for the IoT Service, any IoT Service Platform which communicates to multiple IoT Devices SHALL avoid synchronized behaviour and employ a randomized pattern for accessing the IoT Devices within the IoT Service Platform's domain.

TS.34_6.0_REQ_002	If the (U)SIM subscription associated with an IoT Device is to be placed in a temporarily inactive state (i.e. the subscription is to be disabled for a fixed period of time), the IoT Service Provider SHALL first ensure that the IoT Device is temporarily disabled to restrict the device from trying to register to the network once the SIM is disabled. Before the (U)SIM subscription associated with an IoT Device is changed to a permanently terminated state, the IoT Service Provider SHALL ensure that the IoT Device is permanently disabled to stop the device from trying to register to the network once the SIM is permanently disabled. Note: The IoT Service Provider SHOULD carefully consider permanently terminating IOT devices which are not easily serviceable as it would require manual intervention (i.e. a service call) to re-enable the IoT Device.
TS.34_6.0_REQ_003	If the IoT Service Platform uses SMS triggers to wake up its IoT Devices, the IoT Service Platform SHOULD avoid sending multiple SMS triggers when no response is received within a certain time period.
TS.34_6.0_REQ_004	The IoT Service Platform SHOULD be aware of the state of the IoT Device and only send 'wake up' triggers when the IoT Device is known to be attached to the mobile network.
TS.34_6.0_REQ_005	The IoT Service Platform SHOULD authenticate the IoT Device prior to data communication. The strength of authentication used SHOULD be appropriate to the IoT Service.

7 Policy-based Connection Efficiency Requirements (Normative Section)

7.1 Introduction

In order to complement basic connection efficiency mechanisms such as defined in sections 8 and 9, the IoT Communication Module may implement a flexible policy-based solution whiose primary characteristics are:

1) To take into account the kind of data/communication that is requested by the Device Application ("regular", "urgent", "low priority", as categorized by the Application).

2) To be based on a set of policies that can be controlled and reconfigured over the air using standardized device management mechanisms.

This Policy-based solution has two main components: the policy which defines the intended interaction of the IoT Communication Module with the mobile network and a policy enforcement engine within the IoT Communication Module.

Policy: A policy is defined by the MNO based on its network's specific access preferences. The defined policy can be assigned or changed via remote mechanisms (e.g. OMA-DM, SIM-OTA), local mechanisms (e.g. via AT cmd), or set at the factory (i.e. a default policy). The policy is built up using rules where each rule defines an action (e.g. block GRPS Attaches) which is to be taken by the IoT Communication Module when the rule's conditions

are met (e.g. GMM Error =1). The conditions can be compound expressions based on the IoT Communication Module's current state, as well as counters and timers.

Policy enforcement engine: The policy enforcement engine is code that runs within the IoT Communication Module and is responsible for enforcing the allocated policy. The engine evaluates the rules and executes the actions. Some actions discussed include: Blocking IMSI attach, GPRS attach, PDP context activation, PDN connectivity procedure and SMS-MO, switching PLMNs, and resetting the Communication Module. Some rule conditions discussed include: counting IMSI attaches, GPRS attaches, PDP context activations, PDN connectivity procedure and SMS-MO's and their associated errors.

Example Rule: The following example rules has the IoT Communication Module block GPRS Attaches after a GMM Error codes: x,y,or z is received and then initially back-offs between 10-20 minutes (i.e. IoT Communication Module randomizes in this range), then between 20-30 minutes, then 30-40 minutes thereafter:

Action: [Block] [GPRS Attaches]

Condition: When [GMM errors] [x,y,z] [>=] [1] in [10-20,20-30,30-40] mins

7.2 Policy-based mechanism requirements

7.2.1 General mechanism

TS.34_7.2.1_REQ_001	A IoT Communication Module can have many Connection Efficiency Policies configured, but only one of these CE Policies SHALL be active at any given time.
TS.34_7.2.1_REQ_002	A Connection Efficiency Policy (CE Policy) SHALL contain both: • A set of 1 to 31 arbitrarily defined Connection Efficiency Service Classes corresponding to desired service levels • A set of 1 to 127 Connection Efficiency Rules that govern the behaviour of the IoT Communication Module
TS.34_7.2.1_REQ_003	Individual Connection Efficiency Rules SHALL each contain all of: • A set of 1 to 15 conditions related to the status of the IoT Communication Module (e.g. Time of the day, IoT Device location related to cell, IMSI, ICCID, IMEI) • A set of 0 to 31 applicable Service Classes for that particular Rule. • An individual action such as blocking traffic, retrying a connection, switching network, …

7.2.2 Connection Efficiency Policy Management

TS.34_7.2.2_REQ_001	Connection Efficiency Policies SHALL be manageable remotely using a secured OMA Lightweight M2M connection that allows the management of dedicated OMA LwM2M objects.
TS.34_7.2.2_REQ_002	Connection Efficiency Policies MAY be managed locally using secured (password-protected) AT commands.

7.3 Example application: Connect IoT Device with back-off procedure

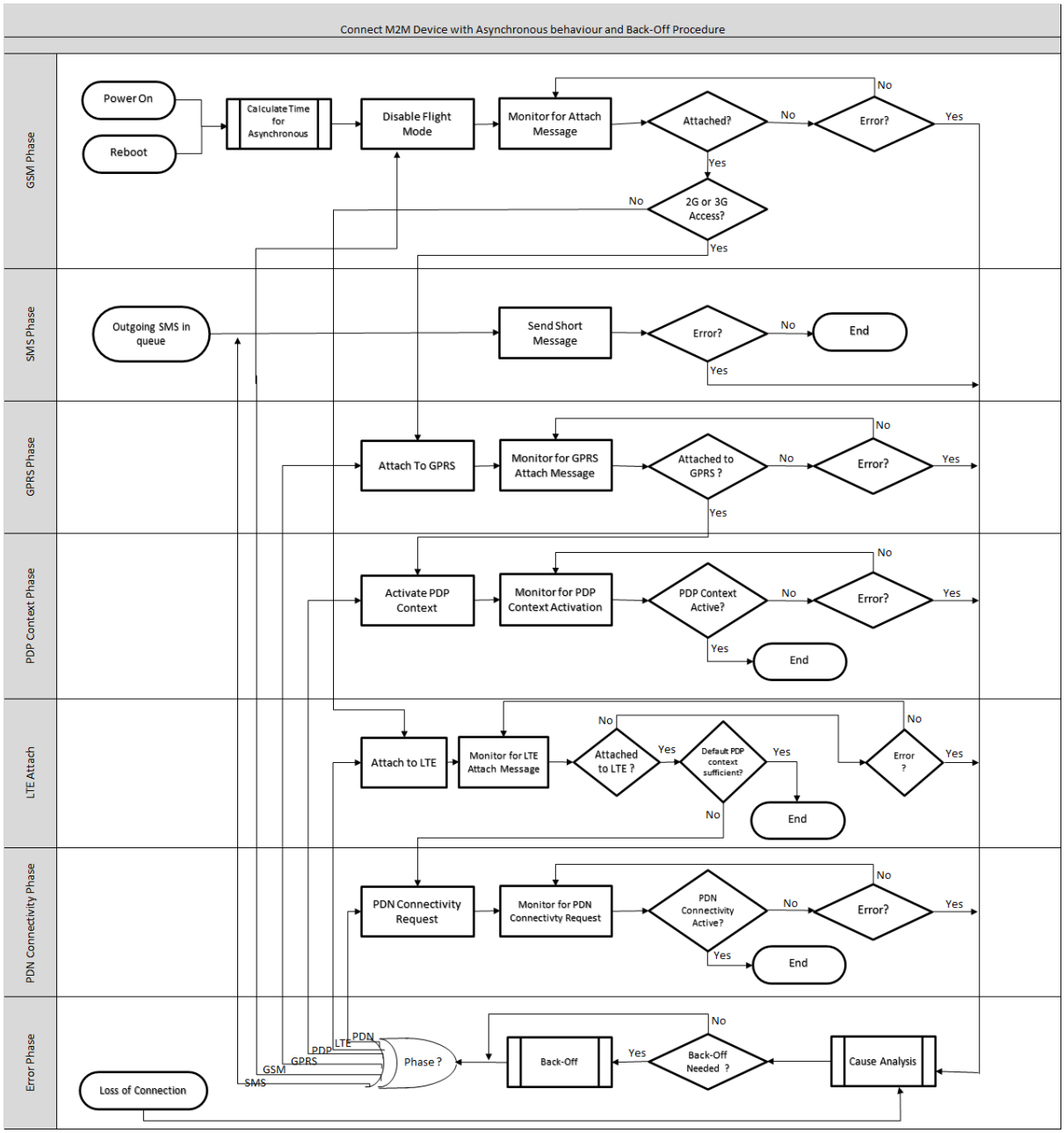


Figure 3: Example Logic Flow for Back Off Procedure

8 Radio Policy Manager Requirements (Normative Section)

This section contains a set of non-standardised features which, when implemented within the IoT Communications Module, will help protect the mobile network from signalling overload.

These features come as a complement to other requirements specified in this document, in particular the ones on the IoT Device Application in section 4.

8.1 Overview

Radio Policy Manager (RPM) objectives are as follows:

- Protect the Network by performing “Connection Aggression Management” which is necessary when the device is aggressively trying to access the network following various NAS reject scenarios
- Enhance Device Operation by making sure the device is back to normal operating mode following a network failure/reject scenario.

8.2 Radio Policy Manager Requirements

8.2.1 General

TS.34_8.2.1_REQ_001	**Default RPM Parameter Settings** The Radio Baseband Chipset SHALL use default RPM parameter settings (see TS.34_8.2.4_REQ_010) saved within the IoT Communication Module Firmware when RPM parameters are not present on UICC.
TS.34_8.2.1_REQ_002	**RPM Activation Control – (U)SIM Present, RPM Parameters Present** If the UICC contains RPM parameters, RPM functionality SHALL be enabled/disabled within the Radio Baseband Chipset based on the setting of the parameter “RPM_ Flag” on the UICC. Note*:* UICC based RPM parameters, if present, SHALL take precedence over any IoT Communication Module Firmware based RPM parameters.
TS.34_8.2.1_REQ_003	**RPM Activation Control – (U)SIM Present, RPM Parameters Not Present** If the UICC does not contain the RPM parameters, RPM functionality SHALL be enabled/disabled based on the default setting of the parameter “RPM_ Flag” saved within the IoT Communications Module Firmware.
TS.34_8.2.1_REQ_004	**RPM Activation Control - Roaming Status** The enabling/disabling of RPM functionality within the Radio Baseband Chipset SHALL be independent of whether the IoT Device is roaming or not.
TS.34_8.2.1_REQ_005	**RPM Parameter Reconfiguration** All RPM parameters SHALL be reconfigurable as per TS.34_8.2.4_REQ_007 “RPM Parameters Remote Management”, TS.34_8.2.4_REQ_008 “RPM (U)SIM Parameters” and TS.34_8.2.4_REQ_009 ”RPM (U)SIM Parameter Updates”.

TS.34_8.2.1_REQ_006	**RPM Version Implemented** At each power up, the Radio Baseband Chipset SHALL check if "RPM version Implemented" on the module is different from the file "EF-RPM Version Implemented" on the (U)SIM card. The update to this file SHOULD be done as early as possible in the power up process. The current version of RPM specified in this document is 2, (i.e. the Radio Baseband Chipset will write 2 to file "EF-RPM Version Implemented" if its implementation of RPM complies with this version of the requirement and is different of the value in (U)SIM card). The version number will be updated when new version of RPM requirement is released.

8.2.2 Mobility Management

TS.34_8.2.2_REQ_001	**RPM Operation Management Counters** RPM Operation Management Counters (C-BR-1, C-R-1, and C-PDP-1 to C-PDP-4) are used to assist monitoring and debugging RPM operation issues. These counters are saved in the (U)SIM. Functionalities related to RPM Operation Management Counters SHALL be disabled if RPM parameters are not present on the (U)SIM.
TS.34_8.2.2_REQ_002	**Reset RPM Operation Management Counters** All RPM Operation Management counters SHALL be reset to 0 if "RPM parameters" or "RPM Operational Management Counters Leak Rate" is updated by OTA. Note: This can be determined from the FILE LIST TLV object in the REFRESH command.
TS.34_8.2.2_REQ_003	**Control Number of Reset** In permanent MM/GMM/EMM reject scenarios described in TS.34_8.2.2_REQ_006 "Handling of "Permanent" MM/GMM/EMM Reject", RPM SHALL allow up to N1 application initiated software resets per hour. This requirement SHALL be disabled if N1 is set to 0. Note: RPM initiated resets SHALL be excluded from N1. User initiated hardware resets SHALL always be allowed and excluded from N1. EMM Reject causes are only applicable to E-UTRAN capable devices.
TS.34_8.2.2_REQ_004	**Increment Counter C-BR-1** RPM SHALL increment counter C-BR-1 by 1 for every reset that it denies access to the mobile network triggered by requirement TS.34_8.2.2_REQ_003. The counter SHALL not roll over. (i.e. 0xFF+1=0xFF).
TS.34_8.2.2_REQ_005	**Reset Counter/timer Related to N1** UE internal counter/timer related to N1 SHALL be reset when UE successfully registers on CS & PS domain. C-BR-1 SHALL not be reset.
TS.34_8.2.2_REQ_006	**Handling of "Permanent" MM/GMM/EMM Reject** RPM SHALL wait for time T1 (or T1 ext) and reset the Radio Baseband Chipset when the following "permanent" MM/GMM/EMM reject causes are received: • MM Reject Cause # 2 (IMSI Unknown in HLR) • MM Reject Cause # 3 (Illegal MS) • MM Reject Cause # 6 (Illegal ME)

	• GMM Reject Cause # 2 (IMSI Unknown in HLR) • GMM Reject Cause # 3 (Illegal MS) • GMM Reject Cause # 6 (Illegal ME) • GMM Reject Cause # 7 (GPRS Services not allowed) • GMM Reject Cause # 8 (GPRS Services and Non-GPRS Services not allowed) • EMM Reject Cause #2 (IMSI unknown in HSS) • EMM Reject Cause #3 (Illegal UE) • EMM Reject Cause #6 (Illegal ME) • EMM Reject Cause #8 (EPS services and non-EPS services not allowed) • EMM Reject Cause #7 (EPS services not allowed) This requirement SHALL be disabled if T1 is set to 0. Note: Timer SHALL not be re-started if an instance of the same timer is already running. EMM Reject causes are only applicable to E-UTRAN capable Radio Baseband Chipsets.
TS.34_8.2.2_REQ_007	**Increment Counter C-R-1** RPM SHALL increment counter C-R-1 by 1 when reset is triggered by T1 or T1 ext. The counter SHALL not roll over.
TS.34_8.2.2_REQ_008	**Stop Timer Related to T1 or T1 ext** UE internal timer related to T1 (T1 ext) SHALL be stopped when UE is reset (hardware or software). In other words, RPM SHALL not reset the Radio Baseband Chipset if it is already reset by the IoT Device Application or IoT Communications Modem.
TS.34_8.2.2_REQ_009	**Handling of Location Update Ignore** If Location Update Request is ignored by network, RPM SHALL ensure that any PS related service request from IoT Device Application will not trigger additional Location Update Request on top of requests that would have been sent by the IoT Communications Module without the service request.
TS.34_8.2.2_REQ_010	**Handling of Attach Request Ignore** If Attach Request is ignored by network, RPM SHALL ensure service request from IoT Device Application will not trigger additional Attach Request on top of requests that would have been sent by IoT Communications Module without the service request.

8.2.3 Session Management

TS.34_8.2.3_REQ_001	**Handling of PDP Context Activation Request / PDN Connectivity Request Ignore** If RPM determines that a PDP Context Activation Request / PDN Connectivity Request has been ignored by the network, RPM SHALL use back-off algorithm to ensure that no more than F1 PDP Context Activation Requests / PDN Connectivity Requests are sent to the same APN every hour. See TS.34_8.2.3_REQ_007 “Minimum Requirement of Back-off Algorithm in PDP Context Activation Reject/Ignore Scenarios” for the minimum requirement of the back-off algorithm. This requirement SHALL be disabled if F1 is set to 0.

TS.34_8.2.3_REQ_002	**Increment Counter C-PDP-1** RPM SHALL increment counter C-PDP-1 by 1 when PDP Context Activation Request / PDN Connectivity Request is ignored by RPM because of requirement TS.34_8.2.3_REQ_001 “Handling of PDP Context Activation Request / PDN Connectivity Request Ignore”. The counter SHALL not roll over.
TS.34_8.2.3_REQ_003	**Handling “Permanent” SM Reject Causes** If PDP Context Activation / PDN Connectivity Request is rejected with the following “permanent” reject causes: • #8 (Operator Determined Barring) • #27 (Missing or Unknown APN) • #28 (Unknown PDP Address or PDP type) • #29 (User Authentication Failed) • #30 (Activation Rejected by GGSN) • #32 (Service Option Not Supported) • #33 (Requested Service Option Not Subscribed) RPM SHALL use back-off algorithm to ensure that no more than F2 PDP Context Activation Requests / PDN Connectivity Requests are sent to the same APN every hour. See TS.34_8.2.3_REQ_007 “Minimum Requirement of Back-off Algorithm in PDP Context Activation / PDN Connectivity Request Reject/Ignore Scenarios” for the minimum requirement of the back-off algorithm. This requirement SHALL be disabled if F2 is set to 0.
TS.34_8.2.3_REQ_004	**Increment Counter C-PDP-2** RPM SHALL increment counter C-PDP-2 by 1 when PDP Context Activation Request / PDN Connectivity Request is ignored by RPM because of requirement TS.34_8.2.3_REQ_003 “Handling “Permanent” SM Reject Causes”. The counter SHALL not roll over.
TS.34_8.2.3_REQ_005	**Handling “Temporary” SM Reject Causes** If PDP Context Activation / PDN Connectivity Request is rejected with the following “temporary” reject causes: • #25 (LLC or SNDCP failure) • #26 (Insufficient resources) • #31 (Activation Rejected, Unspecified) • #34 (Service option temporarily out of order) • #35 (NSAPI already used) • #38 (Network failure) • #102 (No response, timeout) • #111 (Protocol error, unspecified) RPM SHALL use back-off algorithm to ensure that no more than F3 PDP Context Activation / PDN Connectivity Requests are sent to the same APN every hour. See TS.34_8.2.3_REQ_007 “Minimum Requirement of Back-off Algorithm in PDP Context Activation Reject/Ignore Scenarios” for the minimum requirement of the back-off algorithm. This requirement SHALL be disabled if F3 is set to 0.

TS.34_8.2.3_REQ_006	**Increment Counter C-PDP-3** RPM SHALL increment counter C-PDP-3 by 1 when PDP Context Activation Request / PDN Connectivity Request is ignored by RPM because of requirement TS.34_8.2.3_REQ_005 “Handling “Temporary” SM Reject Causes”. The counter SHALL not roll over.
TS.34_8.2.3_REQ_007	**Minimum Requirement of Back-off Algorithm in PDP Context Activation / PDN Connectivity Reject/Ignore Scenarios** The back-off algorithm SHALL be used to ensure that no more than Fx PDP Context Activation / PDN Connectivity Requests are sent to the same APN within 1 hour. Assuming enough PDP Context Activation / PDN Connectivity Requests are received by RPM, the back-off algorithm SHALL allow at least MAX (0.05*Fx, 1) of PDP Context Activation / PDN Connectivity Requests to be sent to the same APN within each 15 minutes window. Ideally the back-off algorithm will come to a steady state after 1 hour. The goal of the algorithm is to avoid excessive number of network connection attempts within short timeframe and at the same time to allow reasonable number of network connection attempts to pass through in order to restore service. This is especially important for IoT Devices that are deployed remotely without easy access by the End Customer or the Mobile Network Operator. Note*:* Fx is the upper limit for the number of requests that the back-off algorithm SHOULD allow in an hour. It is OK (more desirable) if the actual number of requests allowed is less than that.
TS.34_8.2.3_REQ_008	**PDP Context or PDN Connectivity Activation/Deactivation Management** RPM SHALL allow no more than F4 PDP Context Activation Requests / PDN Connectivity Requests each followed by a PDP Context or PDN Connectivity Deactivation Request to be sent to the same APN within one hour (i.e. F4 PDP Context or PDN Connectivity Activation/ Deactivation pairs per hour). After the limit F4 is reached, RPM SHALL ignore subsequent PDP Context Activation Requests / PDN Connectivity Requests to the same APN. This requirement SHALL be disabled if F4 is set to 0.
TS.34_8.2.3_REQ_009	**Increment Counter C-PDP-4** RPM SHALL increment counter C-PDP-4 by 1 when PDP Context Activation Request / PDN Connectivity Request is ignored by RPM because of requirement TS.34_8.2.3_REQ_008 “PDP Context Activation/Deactivation Management”. The counter SHALL not roll over.

8.2.4 Timers and Counters

TS.34_8.2.4_REQ_001	**RPM Timer Values** Value of RPM parameter T1 SHALL be within a time window of [-10%, +10%] of the average value specified in default parameters stored in the IoT Communications Module and on the (U)SIM card.
TS.34_8.2.4_REQ_002	**Reset Timers/counters In PDP context or PDN Connectivity Reject/ignore Requirements** The Radio Baseband Chipset’s internal timers/counters in PDP reject/ignore requirements SHALL be reset after a PDP context is successfully activated.

TS.34_8.2.4_REQ_003	**RPM Behaviour upon (U)SIM Change** All RPM parameters SHOULD be reset upon UICC change. Determination of UICC change SHOULD be based on ICCID.
TS.34_8.2.4_REQ_004	**Periodic Decrement of RPM Operation Management Counter C-BR-1** If LR-1 is NOT 0, C-BR-1 SHALL be decremented by 1 every LR-1 hours if C-BR-1 is greater than 0. C-BR-1 SHALL never be negative. C-BR-1 SHALL not be decremented if LR-1 is 0.
TS.34_8.2.4_REQ_005	**Periodic Decrement of RPM Operation Management Counter C-R-1** If LR-2 is NOT 0, C-R-1 SHALL be decremented by 1 every LR-2 hours if C-R-1 is greater than 0. C-R-1 SHALL never be negative. CR-1 SHALL not be decremented if LR-2 is 0.
TS.34_8.2.4_REQ_006	**Periodic Decrement of RPM Operation Management Counter C-PDP-1 to C-PDP-4** If LR-3 is NOT 0, C-PDP-1/C-PDP-2/C-PDP-3/C-PDP-4 SHALL be decremented by 1 every LR-3 hours if C-PDP-1/C-PDP-2/CPDP- 3/C-PDP-4 is greater than 0. C-PDP-1/C-PDP-2/C-PDP-3/CPDP- 4 SHALL never be negative. C-PDP-1/C-PDP-2/C-PDP-3/C-PDP-4 SHALL not be decremented if LR-2 is 0
TS.34_8.2.4_REQ_007	**RPM Parameters Remote Management** Mobile Network Operators will use their (U)SIM OTA mechanism to manage RPM parameters remotely. The IoT Communication Module based RPM parameters SHALL not be managed by the Mobile Network Operator.
TS.34_8.2.4_REQ_008	**RPM (U)SIM Parameters** The following RPM parameters SHALL be present on the Mobile Network Operator's (U)SIMs (see TS.34_8.2.1_REQ_002 "RPM Activation Control – (U)SIM Present, RPM Parameters Present") as follows: • DF-ARMED AGENT - 3F00/7F66/5F40 (linked file to ADF (USIM)/7F66/5F40) • EF-RPM Enabled Flag - 3F00/7F66/5F40/4F40 (linked file to ADF (USIM)/7F66/5F40/4F40) • EF-RPM Parameters - 3F00/7F66/5F40/4F41 (linked file to ADF (USIM)/7F66/5F40/4F41) • EF-RPM Operational Management Counters Leak Rate - 3F00/ 7F66/5F40/4F42 (linked file to ADF (USIM)/7F66/5F40/4F42) • EF-RPM Operational Management Counters - 3F00/7F66/5F40/ 4F43 (linked file to ADF (USIM)/7F66/5F40/4F43) • EF-RPM Version Information 3F00/7F66/5F40/4F44 (linked file to ADF (USIM)/7F66/5F40/4F44)
TS.34_8.2.4_REQ_009	**RPM (U)SIM Parameter Updates** If the (U)SIM based RPM parameters are updated via OTA, the (U)SIM SHALL issue a REFRESH command of Refresh Type FILE CHANGE NOTIFICATION and also containing a FILE LIST TLV object. The Radio Baseband Chipset SHALL then re-read the (U)SIM based RPM Parameters and start using the updated parameters. All RPM

<table>
<tr><td></td><td>related counters/timers SHALL be reset after RPM parameters are updated via OTA.</td></tr>
<tr><td>TS.34_8.2.4_REQ_010</td><td>RPM Parameter Default Value
RPM parameter default values are listed below:
<table><tr><th>Name</th><th>Description</th><th>Value</th></tr><tr><td>RPM_Flag</td><td>Indicates whether RPM functionality is to be enabled or disabled at power up</td><td>1 (ON)</td></tr><tr><td>N1</td><td>Max number of SW resets per Hour allowed by RPM following “permanent” MM/GMM/EMM reject</td><td>1</td></tr><tr><td>T1</td><td>Average time before RPM resets modem following permanent MM/GMM/EMM reject</td><td>60 minutes</td></tr><tr><td>T1_ext</td><td>Average time before RPM resets modem following permanent MM/GMM/EMM reject if T1 = 0xFF</td><td>48 hours</td></tr><tr><td>F1</td><td>Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by RPM following a PDP Activation Ignore Scenario</td><td>60</td></tr><tr><td>F2</td><td>Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by RPM following a “Permanent” PDP Activation Reject</td><td>30</td></tr><tr><td>F3</td><td>Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by RPM following a “Temporary” PDP Activation Reject</td><td>60</td></tr><tr><td>F4</td><td>Max number of PDP context or PDN Connectivity Activation/ Deactivation Requests per Hour allowed by RPM</td><td>30</td></tr></table></td></tr>
</table>

8.3 RPM (U)SIM Requirements

8.3.1 EF-RPM Enabled Flag Description

This EF indicates if the RPM functionality on the device is to be enabled or disabled at power up. The file shall reside under the DF-ARMED AGENT on both the SIM and USIM applications.

General File Information

Path	3F00/7F66/5F40/4F40 (this is a linked file to ADF(USIM)/7F66/5F40/4F40)
File Type	Transparent

File Body Size	1 byte
Number of Records	N/A
Record Size	N/A
Invalidated at Personalization?	No
Readable and Updateable When Invalidated?	No
Redundancy in Physical File Implementation (to support high update frequency)?	No

Access Conditions

Operation	**Mode**	
	Local	**Remote (OTA)**
Read	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Update	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Invalidate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Rehabilitate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification

Structure and High Level Contents

Byte	**Parameter**	**Description**	**Contents**
1	RPM Enabled Flag	Indicates whether RPM functionality is to be enabled or disabled at power up	• 0x00 - RPM SHALL be disabled at power up • 0x01 to 0XFF - RPM SHALL be enabled at power up

8.3.2 EF-RPM Parameters

Description

This file contains the RPM parameters that are used for the various scenarios defined in the RPM requirements. The file shall reside under the DF-ARMED AGENT on both the SIM and USIM applications.

General File Information

Path	3F00/7F66/5F40/4F41 (this is a linked file to ADF(USIM)/7F66/5F40/4F41)

File Type	Transparent
File Body Size	32 bytes
Number of Records	N/A
Record Size	N/A
Invalidated at Personalization?	No
Readable and Updateable When Invalidated?	No
Redundancy in Physical File Implementation (to support high update frequency)?	No

Access Conditions

Operation	**Mode**	
	Local	**Remote (OTA)**
Read	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Update	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Invalidate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Rehabilitate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification

Structure and High Level Contents

Byte	**Parameter**	**Description**	**Contents**
1	N1	Max number of SW resets per Hour allowed by RPM following “permanent” MM/GMM/EMM reject	0x00 – The requirement is disabled 0x01 to 0xFF - defines the number of resets per hour
2	T1	Average time before RPM resets modem following permanent MM/GMM/EMM reject	0x00 – The requirement is disabled 0x01 to 0xFE - defines in 6 min increments the time to reset after receiving a permanent MM/GMM/EMM reject, i.e. MM#2 0xFF - Timer value to be considered is T1_ext
3	F1	Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by RPM following a PDP Activation Ignore Scenario	0x00 – The requirement is disabled 0x01 to 0xFF – The max attempts allowed
4	F2	Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by	0x00 – The requirement is disabled 0x01 to 0xFF – The max attempts allowed

		RPM following a “Permanent” PDP Activation Reject	
5	F3	Max number of PDP Activation Requests / PDN Connectivity Requests per Hour allowed by RPM following a “Temporary” PDP Activation Reject	0x00 – The requirement is disabled 0x01 to 0xFF – The max attempts allowed
6	F4	Max number of PDP Context or PDN Connectivity Activation/Deactivation Requests per Hour allowed by RPM	0x00 – The requirement is disabled 0x01 to 0xFF – The max attempts allowed
7	T1_ext	Average time before RPM resets modem following permanent MM/GMM/EMM reject if T1 = 0xFF	0x00 – The requirement is disabled 0x01 to 0xFE - defines in 1 hour increments the time to reset after receiving a permanent MM/GMM/EMM reject, i.e. MM#2
8-32	RFU	Reserved for Future Use	Set to 0x00

Note**:** All other values are reserved

8.3.3 EF-RPM Operational Management Counters Leak Rate

Description

This file contains the leak rate for RPM operation management counters. The file shall reside under the DF-ARMED AGENT on both the SIM and USIM applications.

General File Information

Path	3F00/7F66/5F40/4F42 (this is a linked file to ADF(USIM)/7F66/5F40/4F42)
File Type	Transparent
File Body Size	6 bytes
Number of Records	N/A
Record Size	N/A
Invalidated at Personalization?	No
Readable and Updateable When Invalidated?	No
Redundancy in Physical File Implementation (to support high update frequency)?	No

Access Conditions

Operation	**Mode**	
	Local	**Remote (OTA)**
Read	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Update	ADM1	Requires 3GPP TS 31.115 Message

		Integrity Verification
Invalidate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Rehabilitate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification

Structure and High Level Contents

Byte	Parameter	Description	Contents
1	LR-1	Leak rate for C-BR-1	0x00 - C-BR-1 SHALL not be decremented 0x01 to 0xFF - defines number of hours before C-BR-1 is decremented by 1.
2	LR-2	Leak rate for C-R-1	0x00 - C-R-1 SHALL not be decremented 0x01 to 0xFF - defines number of hours before C-R-1 is decremented by 1.
3	LR-3	Leak rate for CPDP-1 to C-PDP-4	0x00 - C-PDP-1 TO C-PDP-4 SHALL not be decremented 0x01 to 0xFF - defines number of hours before C-PDP-1 TO C-PDP-4 is decremented by 1.
4-6	RFU	Reserved for Future Use	Set to 0x00

8.3.4 EF-RPM Operational Management Counters

Description

This file contains the RPM operation management counters that are used to assist monitoring and debugging RPM operation issues. The file shall reside under the DF-ARMED AGENT on both the SIM and USIM applications.

General File Information

Path	3F00/7F66/5F40/4F43 (this is a linked file to ADF(USIM)/7F66/5F40/4F43)
File Type	Transparent
File Body Size	32 bytes
Number of Records	N/A
Record Size	N/A
Invalidated at Personalization?	No
Readable and Updateable When Invalidated?	No
Redundancy in Physical File Implementation (to support high update frequency)?	No

Access Conditions

Operation	Mode	
	Local	**Remote (OTA)**
Read	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Update	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Invalidate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Rehabilitate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification

Structure and High Level Contents

Byte	Parameter	Description	Contents
1	C-BR-1	Counter related to N1	0x00 to 0Xff. Indicate number of control actions triggered by N1.
2	C-R-1	Counter related to T1	0x00 to 0xFF. Indicate number of control actions triggered by T1.
3	C-PDP-1	Counter related to F1	0x00 to 0xFF. Indicate number of control actions triggered by F1.
4	C-PDP-2	Counter related to F2	0x00 to 0xFF. Indicate number of control actions triggered by F2.
5	C-PDP3	Counter related to F3	0x00 to 0xFF. Indicate number of control actions triggered by F3.
6	C-PDP-4	Counter related to F4	0x00 to 0xFF. Indicate number of control actions triggered by F4.
7-32	RFU	Reserved for Future Use	Set to 0x00

8.3.5 EF-RPM Version Implemented

Description

This EF contains the version of RPM that has been implemented and shall be updated by the IoT Device on each power up. The file shall reside under the DF-ARMED AGENT on both the SIM and USIM applications.

General File Information

Path	3F00/7F66/5F40/4F44 (this is a linked file to ADF(USIM)/7F66/5F40/4F44)
File Type	Transparent
File Body Size	1 byte
Number of Records	N/A
Record Size	N/A

Invalidated at Personalization?	No
Readable and Updateable When Invalidated?	No
Redundancy in Physical File Implementation (to support high update frequency)?	No

Access Conditions

Operation	**Mode**	
	Local	**Remote (OTA)**
Read	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Update	ALWAYS	Requires 3GPP TS 31.115 Message Integrity Verification
Invalidate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification
Rehabilitate	ADM1	Requires 3GPP TS 31.115 Message Integrity Verification

Structure and High Level Contents

Byte	**Parameter**	**Description**	**Contents**
1	RPM Version Information	Indicates the version of RPM implemented in the device	0x00 - No Version Information 0x01 - Version 1 0x02 - Version 2 0x03 - Version 3 .. .. 0xFF - Version 255

9 3GPP Connection Efficiency Features (Normative Section)

3GPP provides a number of features to protect mobile networks' from excessive signalling from large numbers of devices in two principle situations:

1. When an IoT Service (associated many IoT Devices) causes a large number of IoT Devices to communicate over a mobile network at the same time; and/or
2. When many IoT Devices are roamers and their serving network fails, then they all attempt move onto a local competing network, and potentially overload this network.

The 3GPP connection efficiency features and their associated IoT Device requirements are described in this section.

It should be noted that both the IoT Device and the Mobile Network must implement these 3GPP features for them to be of benefit to the IoT Service Provider and Mobile Network Operator.

9.1 Rejection of IoT Device Requests with Back-off Timer

When performing mobility management procedures (e.g. location update or routing area update procedures), or session management procedures (e.g. PDP context activation) the mobile network can reject the IoT Device's request with a back-off timer to the device, so that the IoT Device does not re-attempt the request for the specific period of time indicated in the back-off timer.

In 3GPP TS23.401 and TS23.060, two different types of control for the back-off timer are available:

1. APN based congestion control: The network may reject the Session Management requests (e.g. Activate PDP Context Request, PDN Connectivity Request) it receives from devices to a certain APN. This can help the operator to control the amount of traffic using a specific APN.
2. Mobility management congestion control: The network may reject Mobility Management requests (e.g. Attach Request, Routing Area Update, Tracking Area Update) from IoT Devices.

TS.34_9.1_REQ_001	**Congestion Control** The IoT Device SHALL support both APN based congestion control and mobility management congestion control

9.2 Handling of Low Access Priority Indicator

3GPP Release 10 introduces the concept Low Access Priority Indicator (LAPI). The operator can set LAPI in "low priority" IoT Devices, where the application(s) can tolerate longer access delays. The LAPI can be used by the network to reject such an IoT Device from access, and assign a back-off timer preventing the device from immediately repeating the access attempt.

3GPP Release 10 provides an Extended Wait Timer which provides the ability for the mobile network to reject a request with a longer back-off timer than was defined in previous 3GPP releases.

TS.34_9.2_REQ_001	**Low Access Priority Indicator** The IoT Device SHALL support Low Access Priority Indicated (LAPI) and Extended Wait Timer
TS.34_9.2_REQ_002	**SIM–provided Low Access Priority Indicator** If the EFNAS Config Elementary File is present on SIM Card and the LAPI field is filled, then the module SHALL comply with the LAPI value contained in that the SIM EFNAS Config file, meaning that the LAP Indicator is conveyed in RRC and NAS level messages to the network
TS.34_9.2_REQ_003	**Default Value for Low Access Priority Indicator** If the EFNAS Config Elementary File is not present on SIM Card or the LAPI field is not filled, the module SHOULD use an internal default value of LAPI/delay tolerance to be conveyed in RRC and NAS level messages to the network.

TS.34_9.2_REQ_004	**Modification of Default Value for Low Access Priority Indicator** If the module supports an internal default value of LAPI/delay tolerance as per requirement TS.34_9.2_REQ_003, then this internal default value SHOULD be modifiable by the IoT Device Application.

9.3 Implicit Reject in GSM Radio Network

The GSM base transceiver station (BTS) in the serving network can be used to dynamically and quickly control the (over)load from Low Access Priority devices on its RACH, AGCH and SDCCH channels.

TS.34_9.3_REQ_001	**Implicit Reject in GSM Network** Before requesting a signalling channel, an IoT Device that has LAP assigned SHALL check the “Paging” and “Access Grant” broadcast for 20ms. If the BTS has set the “implicit reject” flag (one flag for circuit switched and one flag for packet switched) then the IoT Device SHALL not request a signalling channel, but will back off for a locally generated random period.

9.4 Long Periodic LAU/RAU/TAU

The Periodic Routing Area Update (PRU) and Periodic Tracking Area Update (PTU) timers are used in the Packet Switched domain to control the frequency of PRU and PTU.

In 3GPP Release 10, 3GPP TS23.401 and 3GPP TS23.060 specify that HSS/HLR can be configured with a long PRU/PTU timer per device. During Attach/Routing Area Update/Tracking Area Update procedures, the subscribed PRU/PTU timer values are sent to the SGSN/MME in VPLMN. SGSN/MME then forwards the PRU/PTU timer values to the device.

TS.34_9.4_REQ_001	**Long Periodic LAU/RAU/TAU** The IoT Device SHALL support the extended periodic timers, both for PLU (for circuit switched domain) and PRU/PTU (for packet-switched domain.

9.5 Extended Access Barring

3GPP Release 10 Extended Access Barring (EAB), as specified in 3GPP TS23.060, is a method for a GSM/UMTS network to selectively control access attempts from devices configured for EAB (which are considered more tolerant to access restrictions than other devices) in order to prevent overload of the access network and/or the core network, without the need to introduce any new device access classes.

In the case of congestion, the network could restrict access from IoT Devices configured for EAB while permitting access from other devices. When the network determines that it is appropriate to apply EAB, it broadcasts necessary information on the BCCH to provide EAB control for devices.

TS.34_9.5_REQ_001	**Extended Access Barring** The IoT Device SHALL support Extended Access Barring

9.6 Extended NMO-I

Network Mode of Operation I (NMO-I) enables an IoT Device to perform combined attach towards the packet switched domain. Otherwise, the IoT Device will perform individual attaches to the circuit switched and packet switched domains.

When a large number of roaming IoT Devices attach to a VPLMN, failure of one mobile network might have a domino effect on the other local competing networks, potentially leading to failure of all the networks. The use of combined attach reduces the signalling load on the serving network. However, this might not be beneficial for the operator to apply for all categories of IoT Devices.

Extended NMO-I is introduced in 3GPP Release 10 to allow the mobile network operator to control if a device should perform combined attach or not.

TS.34_9.6_REQ_001	**Extended Network Mode of Operation** The IoT Device SHALL support Extended Access Barring

9.7 Minimum Periodic Search Timer

Pre-“3GPP Release 10” roaming devices do a background search for “more preferred” mobile networks in that country using the timer EF_{HPPLMN} (Higher Priority PLMN search period) which is typically set to 6 or 12 minutes. Consequentially if the most preferred network fails, masses of roaming devices would move to a non-preferred network in that country and, every 6 or 12 minutes attempt (and fail) to return to the preferred network.

The “minimum periodic search timer” is intended to reduce the frequency of this behaviour.

The device shall use the larger of the “minimum periodic search timer” and the value in EF_{HPPLMN} to control its background search for more preferred networks.

TS.34_9.7_REQ_001	**Minimum Periodic Search Timer** The IoT Device SHALL support Minimum periodic timer

9.8 Attach with IMSI Indicator

If this indicator is set when registering with a new mobile network, the device will present its IMSI rather than a temporary identify. This reduces the signalling load on the new network, as it doesn’t have to try and resolve the temporary id and subsequently request the IMSI from the device. This will help a recipient network if it has to manage an incoming ‘avalanche’ of device registrations coming from a failed network.

The disadvantage of setting this parameter is that if the device moves between networks and attaches using the IMSI, then any active PDP context will be torn down. This would also be the case if the device presented an unresolvable TMSI to the new network.

Note: That if the device is moving between equivalent mobile networks (based on the Release 99 equivalent feature) then Attach with IMSI is not invoked.

TS.34_9.8_REQ_001	**Attach with IMSI Indicator** The IoT Device SHALL support Attach with IMSI Indicator

9.9 Timer T3245

The Timer_T3245_Behaviour parameter controls whether timer T3245 is used by the IoT Device. If T3245 is used, then on expiry it causes the device to erase the forbidden network list and to remove any "invalid SIM" setting. The value of T3245 is defined in 3GPP TS 24.008, and is randomly chosen by the device from the range 24 to 48 hours.

The T3245 timer should be used by IoT Devices which are not easy to service. For example, if a smart meter receives a fatal error such as "IMSI unknown" it will add the network to the forbidden list and never connect to it. It is expensive to send a service technician to the smart meter to clear the forbidden network list. Therefore, the T3245 expiry acts as an automated mechanism to flush the forbidden network list, thereby enabling the smart meter to function again.

TS.34_9.9_REQ_001	**Timer T3245** The IoT Device SHALL support Timer T3245

9.10 Configuration of 3GPP Release 10 Connection Efficiency Parameters

Correct operation of the 3GPP Release 10 congestion control mechanisms described above relies on optimal configuration of the device and/or subscription parameters by the mobile network operator.

TS.34_9.10_REQ_001	**OMA DM 3GPP Configuration** Re-configure the terminal's NAS configuration Management Object (MO), see 3GPP TS 24.368
TS.34_9.10_REQ_002	**SIM OTA 3GPP Configuration** Configure the USIM's file EFNASCONFIG (Non Access Stratum Configuration), see 3GPP TS 31.102

Note: That if both USIM and OMA DM values are present within the IoT Device, 3GPP have specified that the USIM values take precedence (see TS 22.368 section 7.1.1, and TS 31.102 section 4.2.94).

9.11 Power Saving Mode

Power Saving Mode is similar to powering-off the device, but the mobile device that uses PSM remains registered with the network so there is no need to re-attach or re-establish the network connection when the device starts transmitting or receiving data.

TS.34_9.11_REQ_001	**Power Saving Mode** The IoT Communications Module SHOULD support Power Saving Mode as defined in 3GPP TS 23.682 Release 12 to enable an IoT Device connected to an LTE network to reduce its power consumption and network signalling

Annex A Connection Efficiency Use Cases (Informative Section)

Proof of the impact of inefficient IoT Devices can be seen today. The following cases were recently experienced by GSMA Mobile Network Operator members and highlight why the requirements defined within this document are necessary:

A.1 Use of Unintelligent Error Handing Mechanisms

In this case, one of the Mobile Network Operator's customers had an installed base of approx. 375,000 geographically fixed IoT Devices (for use in the homes of consumers). These devices were located in 6 different European markets and the devices normally communicated via fixed line Ethernet connections. In normal circumstances the IoT Devices periodically communicate with the customer's server to report on their status, and these status reports must be acknowledged by the customer's server.

Recently the following sequence of events happened which caused massive disruption and loss of service for a large number of the Mobile Network Operator's customers:

3. On a particular day, the customer's server suddenly and unexpectedly stopped acknowledging the status reports from the IoT devices.
4. The devices treated this as a loss of connectivity over their Ethernet network connections and in an attempt to regain connectivity with the server the IoT Devices all started to 'fall-back' to a GSM/GPRS network connection.
5. All the devices then switched on their GSM Communication Modules and attempted to send status messages via their local GSM/GPRS network but again the acknowledge messages were not received from the server.
6. In this event the devices would reset the GSM Communication Module, forcing it to re-register to the local GSM network and the IoT Devices would try again to contact the server. Eventually all 375,000 devices ended up in an infinite loop with their GSM modems being rebooted every minute or so.
7. As the number of devices which entered this 'reboot' loop grew, the signalling load within the core network of the devices home Mobile Network Operator grew to an unmanageable level. This resulted in one of home network's HLRs became overloaded with registration attempts, which in turn prevented all devices that use (U)SIMs provisioned in that HLR to register to any GSM network.
8. At this point the home Mobile Network Operator as he now has a much wider issue to address. The Mobile Network Operator has to stabilize their core network signalling and in this case the Mobile Network Operator was forced close down major roaming destinations like Germany, France, Austria, Italy, Spain and the UK. This reduced the signalling load, and then each network connection could be re-established one by one to bring the number of devices trying to register to the network back in smaller, more manageable, numbers.

Overall, it took this Mobile Network Operator approximately 48 hours to completely resolve the problem which classified the event as a 'critical' event on their network. If the devices had implemented an intelligent 'back-off' mechanism (intended delivery of the Network efficiency project) when loss of connectivity to the server had been detected then this problem would not have occurred.

A.2 Use of insecure IoT Communications Modules

In this case, the Mobile Network Operator's B2B customer had an installed base of 59 IoT devices used to monitor wind and solar power generation. All of the devices used the same make of IoT Communications Modules.

In December 2013 a sudden increase in calls to Gambia, Latvia, Lithuania, UK and Falkland Islands occurred, all the calls being made by the 59 IoT devices. In total approx. 17,000 calls were made before the Mobile Network Operator discovered the fraud and implemented the necessary countermeasures.

Upon further investigation it was discovered:

- All of the IoT Communications Modules within the IoT Devices had been left configured with default usernames and passwords.
- The hacker had discovered the temporary public IP addresses of the IoT Devices and then logged on to each device using the default username and password.
- The hacker then configured the IoT Communications Modules within the IoT Devices to use dynamic DNS addressing to give each device a permanent IP address.
- The hacker then used these permanent IP addresses to connect to the IoT Devices from the 9th to 15th of December and instruct the devices to make calls.

As a result of this hack, the Mobile Network Operator and its customer incurred a financial cost estimated at 150,000 euros for the ~17,000 illegal calls made by the IoT Devices.

If the IoT device vendor had properly configured the security features provided by the IoT Communications Modules within their IoT Devices this event would not have occurred.

A.3 Radius Server Overload

After an SGSN outage tens of thousands IoT devices that belong to an IoT Service Provider re-register to the GPRS network.

There is no throttling activated on the receiving GGSN, so all requests to activate a PDP Context on the IoT Service Provider's APN is processed.

The APN is configured to authenticate through a RADIUS server hosted by the IoT Service Provider which resides on the remote end of a VPN that terminates in the GGSN.

The RADIUS server is not scaling well and the IoT Service Provider has not added enough resources to the RADIUS server to cater for this peak of authentication requests.

The first thousand requests go through but after that the RADIUS server start to experience problems to respond in a timely manner.

In turn the GGSN resend authentication requests that have timed out, putting even more load on the RADIUS server.

Finally, the RADIUS server's CPU utilization hit 100% and the GGSN starts to suffer from the vast amount of PDP Context activation requests that cannot be authenticated and times out.

The IoT Devices do not have a back-off feature and send new requests to activate PDP Context as soon as the previous times out.

The Mobile Network Operator needs to disable all the IoT Devices' (U)SIMs and re-activate them in batches in order for the RADIUS server to be able to authenticate the requests.

Lessons learned:

- Mobile Network Operators should have a throttling mechanism on GGSNs per APN.
- IoT Application Developers' need to implement a back-off feature for such scenarios.
- IoT Service Providers' back-end engineers must communicate with their organization and request information about active (U)SIMs in order to have the appropriate resources available for RADIUS and back-end systems.

A.4 Fake IMEI case

The existence of IoT devices with fake/incorrect IMEIs presents a problem to the Mobile Network Operator. The problem occurs because there are no regulations to check the IMEIs of devices passing customs clearance and as a result, devices with fake/incorrect IMEIs are easily spreading between different markets without any resistance.

Based on Mobile Network Operator experience there is several typical scenarios of fake/incorrect IMEI:

- Copied IMEI for particular consignment of IoT Devices, where the chip which stores the IMEI was not properly coded by manufacturer.
- Substituted IMEI for the IoT Device, taken from the IMEI range dedicated to different type of device and as a consequence the Network has a misunderstanding of device type.
- Fake IMEI which has been re-flashed by the IoT Device Maker from its original value.

A.5 3GPP Standards Non-compliance Cases

3GPP standards non-compliance has been faced for several devices or even types of devices in signalling flow cases.

Device capabilities which have sent to the Network are different in comparison with real device behaviour, the following cases are most typical:

- False information regarding supported frequencies has been sent to the Network, e.g. GSM 1900 instead of GSM1800
- False information regarding the class of output radio power

These false capabilities stresses the Network and behaves abnormally in terms of Network <-> device interaction.

Incorrect response on technical parameter and requirements which sent by the Network in system information messages:

- Much more often Periodical Location Update independently from Network sent parameters. Ignoring of predefined network parameter of Periodical Location Update interval. Doubled or even tripled signalling load on the Network.

- Frequent reload of the device with related signalling flow such as IMSI attach, GPRS attach which increases Network load. The procedure of reloading mechanism is pre-programmed in device application and could be not optimized to the real Network conditions. E.g. losing of the satellite connections to GPS module of the device could be a criteria for initiation of the device rebooting by its application. It could be a reason for additional network load if car with such device installed could be parked under hangar roof for ex.
- Device inability to make Network attach being sent IMSI attach requests while misunderstanding of Network standard signalling respond which cause devices restart and consequent frequent attach requests.

A.6 Other Reported Examples

- Digital Picture Frame –If the device's cloud based server is not available, the device would start to ping the server every 5 seconds to re-establish network connection. When a Mobile Network Operator has thousands of such devices in their network doing the same exhibiting the same behaviour, it results in a "denial of service" attack.
- M2M Device – When configured with an invalid APN or a deactivated (U)SIM the device still attempts to obtain PDP context at a very aggressive rate, unnecessarily consuming network resources and if deployed on a large scale, would congest or crash the network.
- M2M Device Behaviour after Network Outages – After a network outage, when the network comes back up, a large number of devices will see the network and all attempt to access at the same time. The network is unable to respond to all these simultaneous requests. This puts these devices into a state where they are continually attempting to access and potentially crash the SGSN.

Annex B Connection Efficiency Protection Mechanisms Within Mobile Networks (Informative Section)

Mobile networks operators will implement protection mechanisms within their mobile networks to protect their networks from any harm caused by inefficient IoT Devices and IoT Applications.

This annex lists some of the protection mechanisms that network operators may use, usually as a ‘last resort’, within their networks and describes the impact that such mechanisms may have on the IoT service.

It is recommended that IoT Device makers and IoT Application developers be proactive and implement the requirements listed in this document rather than rely on the network operator’s protection mechanisms. Implementing protection mechanisms within the device will mean the IoT Device maker and IoT Service Provider are best placed to monitor and address device and service performance issues without their services being impacted by Network Operator actions.

B.1 Use of SIM Toolkit Applications

Some operators implement a SIM toolkit application within their SIM card that detects inefficient IoT Device behaviour such as repeated device reboots or aggressive network connection reattempts. If the SIM application detects such behaviour it will temporarily disable the network access credentials within the SIM thus preventing the IoT Device from being able to connect to the network for a period of time. The time period that the SIM disables the network access credentials will increase until the IoT Device behaviour returns to normal.

B.2 Use of Dynamic Billing

Some operators will implement dynamic billing so that IoT customers are subject to different network charges at different times of the day. Such a mechanism could be used, for example, to discourage the mass synchronised behaviour of IoT Devices at certain periods of the day.

B.3 Barring of Network Connectivity

Some operators continuously monitor IoT Device behaviour from within their networks and will temporarily disable the subscriptions associated with IoT Devices if they are creating abnormally high levels of signalling or data load on the network. Network operators will usually apply temporary restrictions for short periods of time until the device behaviour returns to normal. If the IoT Device continues to perform inefficiency, and impacts the overall performance of the network and, potentially, other users of the network, the network operator may permanently disable the subscription associated with the problematic device.

Annex C Advice for IoT Application Developers (Informative Section)

C.1 Bandwidth Awareness and Efficient Network Connection Usage Advice

Special consideration must be taken by IoT developers when developing applications that will communicate over wide area wireless networks because of the fundamental differences in the operation of wire area wireless networks compared to ‘fixed’ wireline networks or local wireless (wireless LAN) networks.

The constraints and limitations of mobile networks should be considered by the developer of an IoT Device Application. Operating within these limitations will potentially result in reduced data upload/download volumes, improved IoT Service reliability and responsiveness, and (if applicable) reduced IoT Device power consumption.

As an example of developer best proactive this section provides advice to IoT Applications Developers who are developing applications that will communicate via 3G networks. Similar considerations should also be applied when developing IoT applications that will communicate using other network technologies (2G, 4G etc.).

Apart from data traffic volume, there are key features in a mobile network that require consideration by the IoT Device Application developer. One such feature within 3G networks is Fast Dormancy, a feature that aims to minimise network signalling and battery consumption, both key issues given the increasing number of IoT Devices.

When an IoT Device Application requests data to be sent or received over a mobile network, the Communications Modem switches from an ‘idle mode’ to a ‘dedicated’ channel state that consumes about 60-100 times more power compared to the ‘idle mode’. In addition to this, the very process of switching from the idle to the dedicated state requires network signalling messages that also take a certain amount of time. Keeping the Communications Modem in a high power state is not ideal as it will both consume network resources and increase the IoT Device’s power consumption.

Between the idle and dedicated channel states there are few more radio resource control (RRC) states that come into use. Fast dormancy technology defines an algorithm that dictates when the IoT Communications Module can be switched to lower state after the last data transmission. Figure 3 below shows how the power drops after a certain period of inactivity in data transfer. Times T1 and T2 are network dependent.

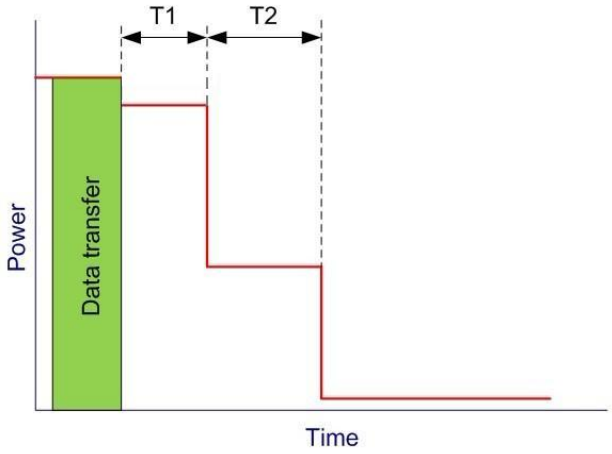


Figure 4: Power Consumption – Example 1

Once the state has switched to idle, establishing a new network connection may require the exchange of between 24-28 signals with the network, which could take one to two seconds.

This is an example of when the app has many short network connections over a specific period of time:

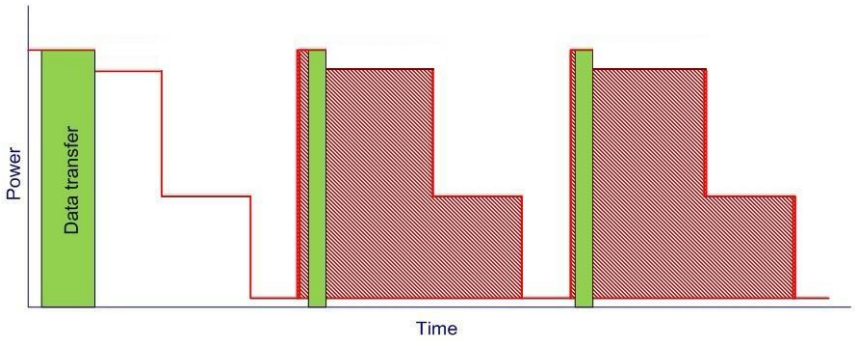


Figure 5: Power Consumption – Example 2

The red-hatched areas in Figure 4 show the overhead in battery usage compared to Figure 5 when all network connections are synchronised and completed in the same time.

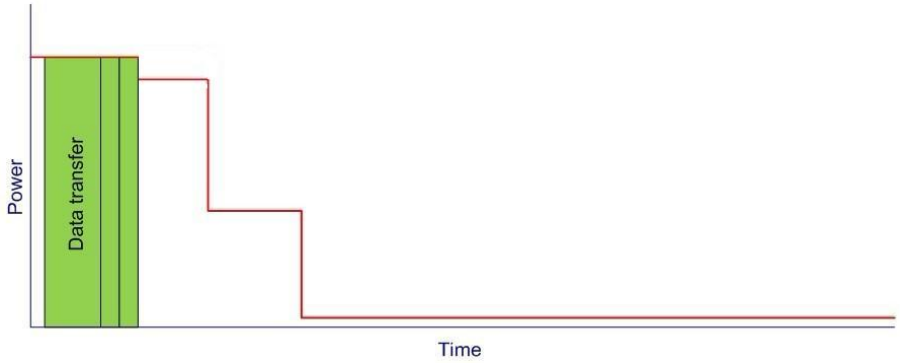


Figure 6: Power Consumption – Example 3

Although most of the timers and conditions of switching between the channel states are network dependent, it is good to at least have an example of the approximate characteristics.

According to tests that have been done by XMPP Foundation:

- Dedicated channel (the highest level) consumes about 380mA. The time before dropping to the lower state is approximately eight seconds
- FACH (shared channel – intermediate level) consumes about 140mA. In order to keep this state and prevent switching into the higher power mode, the packet sizes are recommended to be around 128 bytes and, after deducting TCP and TLS overheads, this leaves only about 70 bytes of actual data. Timeout before switching to the lower state is around eight seconds.

The general recommendation is to transfer data in one go and to not spread out network activities if at all possible.

C.2 IoT Device Application Scaling Advice

IoT Device Applications should be designed to ensure that network activity is not concentrated at specific times and is tolerant of geographical loading problems.

IoT Services are frequently synchronised to a standard clock source and this can result in frequent updates by multiple IoT Devices at exactly the same time (especially for IoT Services that are used by large numbers of End Customers). This can cause overloads to both the IoT Service Platform and the mobile radio network. IoT Services should be designed to spread network activity by different IoT Devices across as wide a time period as possible to reduce such overloads.

To illustrate the point let us take a closer look at example of a IoT Service that checks for service updates periodically (e.g. every 30 minutes), but not necessarily at exact times (e.g. XXhr:00min, XXhr:30min). In such cases, it would be ideal to evenly spread the network activity timings (i.e. the timings which IoT Device Application checks for updates) across devices as in Figure 6 below.

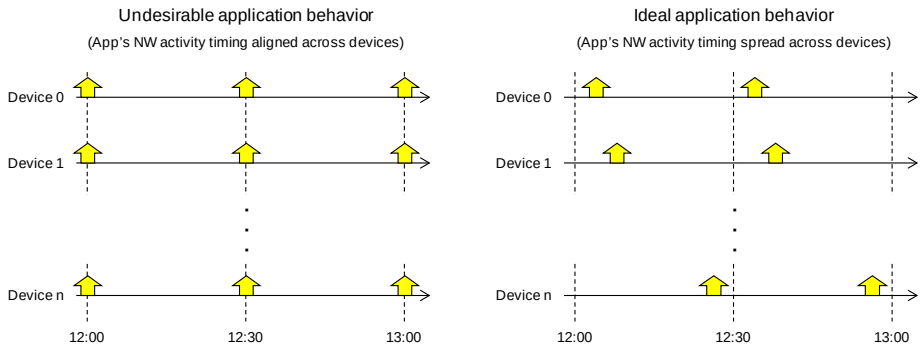


Figure 7: Spreading an IoT Device Application's Network Activity Timing

One way to realise such behaviour would be to schedule network activity timings using relative times (e.g. "30 min from the current time"), and using a timing which would not be aligned across IoT devices as the base timing. For example, the base timing can be the time of the IoT Device boot-up.

Other IoT Services may require data retrieval from servers at exact times of a day (e.g. 05hr:00min, 11hr:00min, 17hr:00min) when the latest information is made available. In such

cases, it would be better to spread the network activity timings (i.e. the timings which the IoT Device Application retrieves data) across IoT Devices within an acceptable time window (e.g. 5min) as in Figure 7 below.

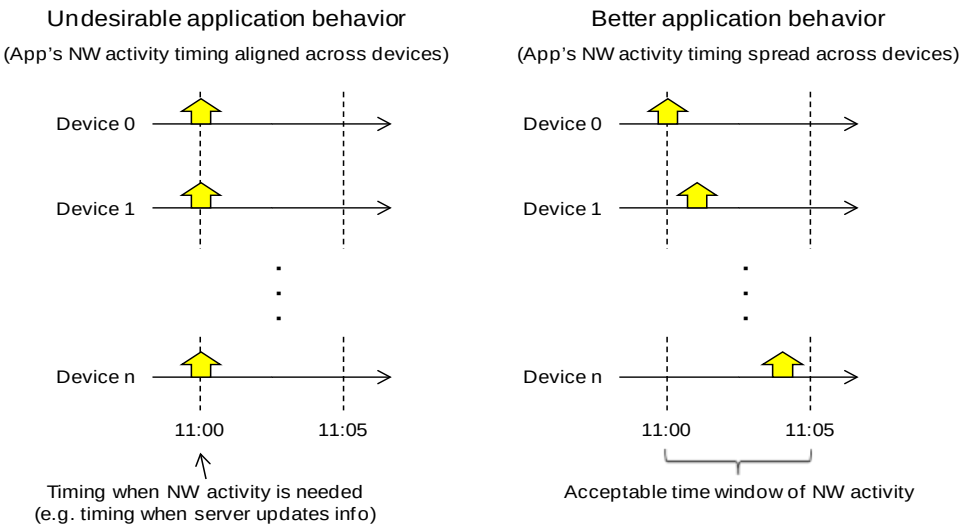


Figure 8: Spreading an Application's Network activity timing within an acceptable window

Such behaviour can be realised by including a random offset (within a desired time window) when scheduling network activities. E.g. "Activity at 17hr:00min + offset", where the offset is defined with a random function having a uniform distribution within the desired window.

IoT Device Application developers are recommended to avoid, as much as possible, using exact times for an IoT Device Application's network activities, and to use randomisation design techniques to spread network activity timings across different IoT Devices. The network capacity of a local area will be significantly lower than the product of the number of IoT Devices and their assigned bandwidth. On occasions there may be large numbers of IoT Devices in a specific location. In general, IoT Device Applications should use some randomisation design techniques to spread network synching and connectivity load.

Annex D Device Diagnostic Requirements (Informative Section)

This section contains requirements which the GSMA intend to further develop and incorporate into the normative section of this document in a future release.

D.1 Remote Diagnostics Recommendations

TS.34_D.1_REQ_001	The IoT Communications Module SHOULD support secure and authenticated OTA protocols to implement the diagnostic requirements stated in RDR2. Examples of related OTA protocols are OMA DiagMon [7], OMA DM [8] and OMA FUMO [9].
TS.34_D.1_REQ_002	The IoT Communications Module SHOULD support the following diagnostic features: • Respond to “ping” query via ICMP; • Report module/device/subscription IDs (IMSI / ICCID / MSISDN); • Report current serving cell ID, received signal level / Received Signal Code Power (RSCP), scrambling code, location area ID; • Report current neighbour cells info; received signal level, ids; • Report the parameters which are related to the network access and applications (i.e. APN, SMSC number, IP, Port); • Report stored history of radio link quality data; • Report circuit-switched call log (mobile-originated and mobile-terminated); • Storage of key events in non-volatile memory then allows the log of these events to be uploaded via TCP/IP; • Start and stop log storage via remote commands; • Attach status (including reason for attach failures); • PDP context status (including reason for context establishment failures); • Report a log of failures (e.g. SMS send failure, software update failure, PIN code failures etc.); • Report hardware/software/firmware versions; • Report status of device integrity check of the HW/SW/configuration files of the IoT Communications Module; • Report status of device integrity check of the HW/SW/configuration files of the host device; • Report battery charge level; • Report packet transfer history statistics (number of Tx, number of Rx, retries); • Report last 5 IP addresses with which the IoT Communications Module communicated; • Report SMS transfer history statistics (i.e. number of Tx, number of Rx, retries); • If IoT Communication Module has location capability, report location; • If IoT Communication Module or host device has a real-time clock capability, report local time; • Upload selected area of IoT Communication Module’s memory (supplied address, length);

	• Download an application to the IoT Communication Module's RAM; • Remove an application in the IoT Communication Module's RAM; • Check status of peripheral devices attached to IoT Communication Module; • Report re-boot history (stored in non-volatile memory); • Report stored history of local servicing of the IoT Communication Module or the host device by technicians (including their ids); • Re-boot IoT Communication Module on remote command; • Report the total amount of memory currently being used and the amount of free memory.

D.2 Local Diagnostic Requirements

TS.34_D.2_REQ_001	The IoT Communications Module SHALL support a local interface (for example RS-232, USB or other interface) over which local diagnostic information may be obtained. The diagnostic interface SHOULD allow: • Manual reboot; • Check of integrity of the h/w, s/w configuration of the IoT Communication Module and/ or the host device; • Display of the cellular environment (including received signal strength, cell ids for serving and neighbour cells); • List of any stored error codes or logs; • Display of selected log; • Display of non-volatile configuration settings; • Capability to test peripherals connected to the IoT Communication Module; • Sending of at and diagnostic commands to the IoT Communication Module; • Check of battery charge status (if applicable).

Annex E GSM/UMTS Cause Code

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
2							IMSI unknown in HLR	This cause is sent to the device if the device is not known (registered) in the HLR. This cause code does not affect operation of the GPRS service, although it MAY be used by a GMM procedure.	The IoT Communications Module SHALL perform a GSM Attach 'Back-off', as defined in section 7 of this document, at next power cycle
	2						IMSI unknown in HLR (NOM1 only)	This cause is sent to the device if the device is not known (registered) in the HLR. This cause code does not affect operation of the GPRS service, although it MAY be used by a GMM procedure.	The IoT Communications Module SHALL perform a GSM and GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
3					103		Illegal device	This cause is sent to the device when the network refuses service to the device either because an identity of the device is not acceptable to the network or because the device does not pass the authentication check, i.e. the SRES received from the device is different from that generated by the network.	The IoT Communications Module SHALL perform a GSM Attach 'Back-off', as defined in section 7 of this document, at next power cycle
	3				106		Illegal device	This cause is sent to the device when the network refuses service to the device either because an identity of the device is not acceptable to the network or because the device does not pass the authentication check, i.e. the SRES received from the device is different from that generated by the network.	The IoT Communications Module SHALL perform a GSM and GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
4							IMSI unknown in VLR	This cause is sent to the device when the given IMSI is not known at the VLR.	As per 3GPP specifications.
5							IMEI not accepted	This cause is sent to the device if the network does not accept emergency call establishment using an IMEI.	The IoT Communications Module SHALL perform the 'Back-off', as defined in section 7 of this document, at next power cycle
6					106		Illegal ME	This cause is sent to the device if the ME used is not acceptable to the network, e.g. blacklisted.	The IoT Communications Module SHALL perform a GSM 'Back-off', as defined in section 7 of this document, at next power cycle

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
	6				106		Illegal ME	This cause is sent to the device if the ME used is not acceptable to the network, e.g. blacklisted.	The IoT Communications Module SHALL perform a GSM and GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
	7				107		GPRS Services Not Allowed	This cause is sent to the device if it requests an IMSI attach for GPRS services, but is not allowed to operate GPRS services.	The IoT Communications Module SHALL perform a GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
8	8						GPRS services and non-GPRS services not allowed	This cause is sent to the device if it requests a combined IMSI attach for GPRS and non-GPRS services, but is not allowed to operate either of them.	The Communications Module SHALL perform a GSM and GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
9	9						Device identity cannot be derived by the network	This cause is sent to the device when the network cannot derive the device's identity from the P-TMSI in case of inter-SGSN routing area update.	The IoT Communications Module SHALL perform a GSM and GPRS Attach 'Back-off', as defined in section 7 of this document, at next power cycle
	10						Implicitly detached	This cause is sent to the device either if the network has implicitly detached the device, e.g. some while after the Mobile reachable timer has expired, or if the GMM context data related to the subscription does not exist in the SGSN e.g. because of a SGSN restart.	As per 3GPP specification
11	11				111		PLMN not allowed	This cause is sent to the device if it requests location updating in a PLMN where the device, by subscription or due to operator determined barring is not allowed to operate.	The IoT Communications Module SHOULD not retry the attach attempt on the same PLMN unless prompted externally to do so (i.e. the IoT Communications Module SHOULD not automatically retry in the same PLMN).
12	12				112		Location Area not allowed	This cause is sent to the device if it requests location updating in a location area where the device, by subscription, is not allowed to operate.	The IoT Communications Module SHOULD not retry the attach attempt on the same LA unless prompted externally to do so (i.e. The IoT Communications Module SHOULD not automatically retry in the same LA).
13	13				113		Roaming not allowed in this location area	This cause is sent to a device which requests location updating in a location area of a PLMN which restricts roaming to that device in that Location Area, by subscription.	The IoT Communications Module SHOULD not retry the attempt on the same LA unless prompted externally to do so (i.e. modem SHOULD not automatically retry in the same LA).

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
	14						GPRS services not allowed in this PLMN	This cause is sent to the device which requests GPRS service in a PLMN which does not offer roaming for GPRS services to that device.	The IoT Communications Module SHOULD not retry the attempt on the same PLMN unless prompted externally to do so (i.e. modem SHOULD not automatically retry in the same PLMN).
15	15						No Suitable Cells In Location Area		The IoT Communications Module SHOULD not retry the attempt on the same cell unless prompted externally to do so (i.e. IoT Communications Module SHOULD not automatically retry in the same cell).
	16						MSC temporarily not reachable (NOM 1 only)	This cause is sent to the device if it requests a combined GPRS attach or routing are updating in a PLMN where the MSC is temporarily not reachable via the GPRS part of the GSM network.	The IoT Communications Module SHALL perform the 'Back-off', as defined in section 7 of this document, at next power cycle
17	17				615		Network failure	This cause is sent to the device if the MSC cannot service a device generated request because of PLMN failures, e.g. problems in MAP.	The IoT Communications Module SHALL perform the 'Back-off', as defined in section 7 of this document, at next power cycle
20	20						MAC failure	This cause is sent to the network if the (U)SIM detects that the MAC in the authentication request message is not fresh	As per 3GPP specifications
21	21						Sync failure	This cause is sent to the network if the (U)SIM detects that the SQN in the authentication request message is out of range	As per 3GPP specifications
22	22				42		Congestion	This cause is sent if the service request cannot be preceded because of congestion (e.g. no channel, facility busy/congested etc.)	The IoT Communications Module SHALL perform the 'Back-off', as defined in section 7 of this document, at next power cycle
23							GSM authentication unacceptable	This cause is sent to the network in UMTS if the MS supports the UMTS authentication algorithm and there is no Authentication Parameter AUTN IE present in the AUTHENTICATION REQUEST message	As per 3GPP specifications
32					132		Service Option Not Supported	This cause is sent when the device requests a service/facility in the CM SERVICE REQUEST message which is not supported by the PLMN.	As per 3GPP specifications

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
33					133		Requested Service Option Not Subscribed	This cause is sent when the device requests a service option for which it has no subscription.	As per 3GPP specifications
34					134		Service option temporarily out of order	This cause is sent when the MSC cannot service the request because of temporary outage of one or more functions required for supporting the service.	The IoT Communications Module SHALL perform the 'Back-off', as defined in section 7 of this document, at next power cycle
38							Call Cannot be identified	This cause is sent when the network cannot identify the call associated with a call re-establishment request.	As per 3GPP specifications
40							No PDP context activated	This cause is sent to the device if the device requests an establishment of the radio access bearers for all active PDP contexts by sending a SERVICE REQUEST message indicating "data" to the network, but the SGSN does not have any active PDP context(s).	As per 3GPP specifications
All other MM codes	All other GMM codes								As per 3GPP specifications
		8					Operator determined barring	This cause indicates that the device has tried to send a mobile originating short message when the device's network operator or service provider has forbidden such transactions.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		26					Insufficient resources	This cause code is used by the device or by the network to indicate that a PDP Context Activation Request / PDN Connectivity Request or PDP Context modification request cannot be accepted due to insufficient resources	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the Iot Communications Module via AT commands.
		27			134		Unknown or missing access point name	This cause code is used by the network to indicate that the requested service was rejected by the external packet data network because the access point name was not included although required or if the access point name could not be resolved.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
		28					Unknown PDP address or PDP type	This cause code is used by the network to indicate that the requested service was rejected by the external packet data network because the PDP address or type could not be recognized.	The IoT Communications Module SHALL perform a GPRS re-attach (i.e. the IoT Communications Module SHALL perform a GPRS detach followed by a GPRS attach)
		29			149		User authentication failed	This cause code is used by the network to indicate that the requested service was rejected by the external packet data network due to a failed user authentication (e.g. rejected by Radius)	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		30					Activation rejected by GGSN	This cause code is used by the network to indicate that the requested service was rejected by the GGSN.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		31					Activation rejected, unspecified	This cause code is used by the network to indicate that the requested service was rejected due to unspecified reasons.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		32			132		Service option not supported	This cause code is used by the network when the device requests a service which is not supported by the PLMN or the APN is invalid.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		33			133		Requested service option not subscribed	This cause is sent when the device requests a service option for which it has no subscription. The difference between this and CMEE 132 is that the network MAY support the requested option, but the user is not subscribed to that option.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands.
		34			134		Service option temporarily out of order	This cause is sent when the MSC\SGSN cannot service the request because of	If a second mobile network is available, the IoT Communications Module SHALL attempt to connect via the alternate mobile network. If no

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
								temporary outage of one or more functions required for supporting the service.	other mobile network is available, the IoT Communications Module SHALL all perform a Back-off, as per section 7 of this document.
		35					NSAPI already used	This cause code is used by the network to indicate that the NSAPI requested by the device in the PDP Context activation is already used by another active PDP Context of this device.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands
		36					Regular PDP Context deactivation	This cause code is used to indicate a regular device or network initiated PDP Context deactivation.	If the IoT Communications Module has not requested the PDP context deactivation it is likely this is due to idle timeout. Immediate reactivation of PDP Context by the IoT Communications Module is OK.
		37					QoS not accepted	This cause code is used by the device if the new QoS cannot be accepted that were indicated by the network in the PDP Context Modification procedure.	As per 3GPP specifications
		38			615		Network Failure	This cause code is used by the network to indicate that the PDP Context deactivation is caused by an error situation in the network.	The IoT Communications Module SHALL perform a Session Management Back-off, as per section 7 of this document, blocking immediately any new service requests sent by to the IoT Communications Module via AT commands
		39					Reactivation requested	This cause code is used by the network to request a PDP Context reactivation after a GGSN restart.	The IoT Communications Module MAY re-establish the PDP Context immediately, but upon failure go to back-off.
		40					Feature not supported	This cause code is used by the device to indicate that the PDP Context activation initiated by the network is not supported by the device.	As per 3GPP specifications
		43					Unknown PDP context	This cause code is used by the network or the device to indicate that the PDP context identified by the Linked TI IE in the secondary PDP Context Activation / PDN Connectivity Request or a network requested secondary PDP context activation is not active.	As per 3GPP specifications

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
		56					Collision with network initiated request	This cause code is used by the network to indicate that the device-initiated request was rejected since the network has requested a secondary PDP context activation for the same service using a network-initiated procedure.	As per 3GPP specifications
		112					APN restriction value incompatible with active PDP context	This cause code is used by the network to indicate that the PDP context(s) or MBMS context(s) have an APN restriction value that is not allowed in combination with a currently active PDP context.	As per 3GPP specifications
			8			8	Operator determined barring	This cause indicates that the device has tried to send a mobile originating short message when the device's network operator or service provider has forbidden such transactions.	The Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the Communications Module via AT commands.
			10			10	Call barred	This cause indicates that the outgoing call barred service applies to the short message service for the called destination.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			21			21	Short message transfer rejected	This cause indicates that the equipment sending this cause does not wish to accept this short message, although it could have accepted the short message since the equipment sending this cause is neither busy nor incompatible.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			22			27	Destination out of service	This cause indicates that the destination indicated by the Device cannot be reached because the interface to the destination is not functioning correctly. The term "not functioning correctly" indicates that a signalling message was unable to be delivered to the remote user; e.g., a physical layer or data link layer failure at the remote user, user equipment off-line, etc.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			28			28	Unidentified subscriber	This cause indicates that the subscriber is not registered in the PLMN (i.e. IMSI not known).	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
									SMS TX requests sent to the IoT Communications Module via AT commands.
			29			29	Facility rejected	This cause indicates that the facility requested by the Device is not supported by the PLMN.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			30			30	Unknown subscriber	This cause indicates that the subscriber is not registered in the HLR (i.e. IMSI or directory number is not allocated to a subscriber).	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			38			38	Network out of order	This cause indicates that the network is not functioning correctly and that the condition is likely to last a relatively long period of time; e.g., immediately reattempting the short message transfer is not likely to be successful.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			41			41	Temporary failure	This cause indicates that the network is not functioning correctly and that the condition is not likely to last a long period of time; e.g., the Device MAY wish to try another short message transfer attempt almost immediately.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			42			42	Congestion	This cause indicates that the short message service cannot be serviced because of high traffic.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			47			47	Resources unavailable, unspecified	This cause is used to report a resource unavailable event only when no other cause applies.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			50			50	Requested facility not subscribed	This cause indicates that the requested short message service could not be provided by the network because the user has not completed the necessary administrative arrangements with its supporting networks.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.

For Communication Module Manufacturers					For IoT Device Application Developers				
MM Cause Code	**GMM cause**	**SM Cause Code**	**RP cause code**	**CP cause code**	**CME ERROR**	**CMS ERROR**	**Cause**	**Reason**	**Proposed action (if different from 3GPP TS 24.008)**
			69			69	Requested facility not implemented	This cause indicates that the network is unable to provide the requested short message service.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
			81			81	Invalid short message transfer reference value	This cause indicates that the equipment sending this cause has received a message with a short message reference which is not currently in use on the MS-network interface.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
				17			Network failure	This cause is sent to the MS if the MSC cannot service an MS generated request because of PLMN failures, e.g. Problems in MAP.	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
				21			Congestion	This cause is sent if the service request cannot be actioned because of congestion (e.g. no channel, facility busy/congested, etc.).	The IoT Communications Module SHALL perform an SMS Back-off, as per section 7 of this document, blocking immediately any new SMS TX requests sent to the IoT Communications Module via AT commands.
					148		Unspecified GPRS error		As per 3GPP specifications

Annex F Example Text to be Inserted Into Contracts and RFQs (Informative Section)

This section contains an example of the text that could be adapted and used as a base for an RFQ or contract between a Mobile Network Operator and IoT Service Provider who would like to connect their IoT Devices to the Mobile Network Operator network. Inserting such text will allow the Mobile Network Operator to reference the key requirements within the GSMA Connection Efficiency Guidelines without having to insert the whole document into their RFQ or contracts.

F.1 Example Text

<<<<<<<<<<<<<<<<<<<<<<<<< CUT HERE >>>>>>>>>>>>>>>>>>>>>>>>>

x.1 Problem Statement

The predicted large scale growth of IoT Devices will create major challenges for Mobile Network Operators. One major challenge that Mobile Network Operators must overcome is the risk caused by the mass deployment of inefficient, insecure or defective IoT Devices on the Mobile Network Operators' networks. When deployed on a mass scale such devices can cause network signalling traffic to increase exponentially which impacts network services for all users of the mobile network. In the worst cases the mass deployment of such IoT Devices can disable a mobile network completely.

IoT Devices overusing the mobile network can affect not only the devices causing the incident but also other devices on the same IoT Service Platform or those devices of other End Customers.

Network signalling resources are dimensioned assuming an overall device usage profile with a sensible balance between traffic and signalling needs. It is therefore important that IoT Devices using mobile networks adhere to some basic principles before they can be safely connected to mobile networks.

Good design is essential to ensure that IoT Device performance is optimized and to prevent failure mechanisms creating runaway situations which may result in network overload.

x.2 Key Words Used to Indicate Requirement Levels

The use of "shall" in this section means that the definition is an absolute requirement of the Mobile Network Operator.

x.3 Definition of Terms

Term	Description
IoT Communications Module	The communications component which provides wide area (2G, 3G, 4G) radio connectivity. Comprising of IoT Communications Module Firmware, Radio Baseband Chipset and UICC
Global Certification Forum (GCF)	An independent worldwide certification scheme for mobile phones and wireless devices that are based on 3GPP standards. For more information, see http://www.globalcertificationforum.org
IoT Device	The combination of both the IoT Device Application and the IoT Communications Module.

Term	Description
IoT Device Application	The application software component of the IoT Device that controls the IoT Communications Module and interacts with an IoT Service Platform via the Communications Module.
IoT Service Provider	The provider of IoT services working in partnership with a Mobile Network Operator to provide an IoT Service to an End Customer. The provider could also be a Mobile Network Operator.
Mobile Network Operator	The mobile network operator(s) connecting the IoT Device Application to the IoT Service Platform.
PTCRB	The independent body established as the wireless device certification forum by North American Mobile Network Operators. For more information, see http://ptcrb.com

x.4 References

Ref	Document Title	Source
1	GSMA IoT Device Connection Efficiency Guidelines	http://www.gsma.com/connectedliving/gsma-iot-device-connection-efficiency-guidelines/

x.5 IoT Service Provider Requirements

The IoT Service Provider's IoT Service Platform shall conform to the requirements stated in Section 6 of the GSMA IoT Device Connection Efficiency Guidelines [1].

The IoT Service Provider shall only connect IoT Devices to the Mobile Operators Network that conform to the requirements stated in the GSMA IoT Device Connection Efficiency Guidelines [1]. Specifically:

1. The IoT Device Application shall conform to all requirements defined in section 4 of the GSMA Connection Efficiency Guidelines [1].
2. The IoT Device's IoT Communication Module shall conform to all requirements defined in section 5 of the GSMA Connection Efficiency Guidelines [1]. Specifically:
   2.1. The IoT Communications Module shall be compliant with 3GPP specifications unless otherwise stated within the GSMA IoT Device Connection Efficiency Guidelines [1].
   2.2. The IoT Communications Module shall be certified by the GCF and/or the PTCRB.
   2.3. The IoT Communications Module shall investigate, and meet as required, the mobile network operator requirements for the target market(s).
   2.4. The IoT Communications Module shall support (dependent upon the target mobile network operator) at least one of the following requirements:
      2.4.1. Radio Policy Manager as defined in section 8 of the GSMA Connection Efficiency Guidelines [1]; OR
      2.4.2. Connection Efficiency requirements as defined in section 7 of the GSMA Connection Efficiency Guidelines [1]; OR
      2.4.3. 3GPP Connection Efficiency features as defined in section 9 of the GSMA Connection Efficiency Guidelines [1].

Note: This option requires the target mobile network operator to have implemented the required 3GPP optional features.

3. If required by the Mobile Network Operator, the IoT Device shall be certified by the GCF and/or the PTCRB.

<<<<<<<<<<<<<<<<<<<<<<<<<<<< CUT HERE >>>>>>>>>>>>>>>>>>>>>>>>>>

Annex G Void

Annex H Document Management

H.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	13-OCT-2014	New PRD CLP.03	PSMC	Ian Smith GSMA
1.1	30-Jan-2014	DHIR requirements moved to normative section of document. Requirements NFM8, NFM9 and NFM10 added. New Annex E New Annex F Editorial Corrections	PSMC	Ian Smith, GSMA
2.0	01-Jul-2015	Updated definitions Updated Generalised IoT Service Architecture Clarifications to DAR3 and NFM7 Requirement DID33 added General editorial corrections	PSMC	Ian Smith, GSMA
	01-Jul-2015	Change of ownership to GSMA TSG and the document number changed to TS.34	TSG	Jerome Hamel, 7Layers
3.0	Feb-2016	Requirements re-numbered New requirements added in section 9.0	TSG	Jerome Hamel, 7Layers
4.0	Nov 2016	New section added, 4.1, 4.2 & 4.3	TSG	Nicolas DAMOUR
5.0	Jan 2018	Updated with approved CR1003	TSG	Jerome Hamel, 7Layers
6.0	April 2020	Updated with approved CR1004	TSG39	Nicolas Damour Sierra Wireless
7.0	Feb 2021	Updated with approved CR1005	TSG#42 ISAG#6	Nicolas Damour

				Sierra Wireless
7.1	June 2021	Updated with approved CR1006	TSG#44	Nicolas Damour Sierra Wireless

Other Information

Type	Description
Document Owner	GSMA TSG
Editor / Company	Nicolas Damour / Sierra Wireless

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at PRD@gsma.com

Your comments or suggestions & questions are always welcome.